10'

1-16

22-27

(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 0 909 037 A1

(12) **EUROPEAN PATENT APPLICATION**
published in accordance with Art. 158(3) EPC

(43) Date of publication:
14.04.1999 Bulletin 1999/15

(21) Application number: 98909841.3

(22) Date of filing: 24.03.1998

(51) Int. Cl.^{6}: **H03M 7/30**, H03M 7/46,
G01R 31/3183

(86) International application number:
PCT/JP98/01273

(87) International publication number:
WO 98/43359 (01.10.1998 Gazette 1998/39)

(84) Designated Contracting States:
DE FR GB

(30) Priority: 24.03.1997 JP 70293/97
24.03.1997 JP 90225/97

(71) Applicant: Advantest Corporation
Nerima-ku, Tokyo 179-0071 (JP)

(72) Inventors:
- ISHIDA, Masahiro
  Sendai-shi, Miyagi 989-3124 (JP)
- YAMAGUCHI, Takahiro
  Sendai-shi, Miyagi 989-3124 (JP)
- TILGNER, Marco
  Sendai-shi, Miyagi 989-3124 (JP)

(74) Representative:
Hoffmann, Eckart, Dipl.-Ing.
Patentanwalt,
Bahnhofstrasse 103
82166 Gräfelfing (DE)

(54) METHOD AND DEVICE FOR COMPRESSING AND EXPANDING DATA PATTERN

(57) A method by which a test pattern to be applied onto an IC for testing can be compressed efficiently. The method comprises determining the number ϕ of data transitions of the pattern at every pin of the IC and the entropy H of the data, distributing the test pattern among a block wherein the ϕ is smaller than a threshold $\phi_M(\phi<\phi_M)$, a block wherein the ϕ is larger than the threshold $\phi_M(\phi>\phi_M)$ and the H is larger than a threshold $H_M(H<H_M)$, and a block wherein the H is larger than the threshold $H_M(H>H_M)$(411), compressing the block wherein $\phi<\phi_M$ by the run-length compressing method, compressing the block wherein $\phi>\phi_M$ and $H<H_M$ by the run-length compressing method after Burrows Wheeler conversion, and compressing the block wherein $H>H_M$ by the LZ compressing method.

Printed by Xerox (UK) Business Services
2.16.7/3.6

Description

TECHNICAL FIELD

[0001] The present invention relates to a method of compressing a data pattern such as a test pattern used for testing, for example, a semiconductor integrated circuit (IC), a method of expanding such data pattern, an apparatus for compressing such data pattern, and an apparatus for expanding such data pattern. In addition, the present invention relates to a compressing method, an expanding method, a compressing apparatus and an expanding apparatus for compressing and expanding a test pattern used for testing a large scale semiconductor integrated circuit (LSI) by each pin basis of a large scale semiconductor integrated circuit (hereinafter referred to as LSI).

BACKGROUND ART

[0002] In an IC testing apparatus (commonly called IC tester) for testing a semiconductor integrated circuit (hereinafter referred to as IC), the data quantity or volume of test patterns applied to an IC to be tested (IC under test) becomes extensive. Therefore, the extensive test pattern data are compressed to be transmitted and the compressed test pattern data are expanded at a receiving side to reconstruct original test pattern data which are applied to an IC under test. As a method for compressing test patterns or other extensive data patterns, a Lempel Ziv's algorithm (LZ compressing method) which utilizes a dictionary, a Huffman's algorithm (Huffman compressing method) which utilizes statistical characteristics of the data, or variations of those various compressing methods have conventionally been used in a data compressing apparatus. Each of those compressing methods is a signal compressing method and each of the data compressing apparatus applies this single compressing method to all the data patterns to be compressed to perform the compression of those data patterns.

[0003] However, since each data pattern such as a test pattern has a considerably different data structure or a statistical characteristic from each other, an efficient compression has not been possible depending on the data pattern to be used. For example, when each of different portions within a same data pattern is significantly different from each other in terms of the pattern structure or the statistical characteristic, a high compression efficiency can be obtainable for a certain portion but the compression efficiency becomes low for remaining portions. As a result, the compression rate is relatively low for the entire data pattern. In addition, regarding the processing time, since the processing time is different depending on the structure of a data pattern, an optimum processing time has not been attained.

[0004] There has been proposed, from this stand point, a data compression wherein a test pattern is divided into blocks each having a different data structure or a statistical characteristic from the others and an appropriate compressing method is applied to each block to compress the data.

[0005] However, since, in this data compressing method, the test pattern to be compressed is compressed in a block basis, it is required that the data expansion is also performed in a block basis. Therefore, it is impossible, in this compressing method, to expand the compressed test pattern in a real time basis.

[0006] Further, talking about the microprocessors produced by Intel Corporation as example, the number of pins has increased year by year as the integration degree is improved as seen in the models 4004 in 1971 having 16 pins, 80286 in 1982 having 68 pins, 80386 in 1985 having 132 pins, 80486 in 1989 having 168 pins, Pentium in 1993 having 296 pins, and Pentium Pro. in 1995 having 387 pins. As a result, an automatic IC test system (ATE) has also been shifted to a system wherein the hardware of a testing apparatus corresponds to a pin (per-pin system). This is because, with the per-pin architecture, the automatic IC testing system can flexibly cope with the increasing number of pins.

[0007] In order to solve the test pattern problems described above, there are required a per-pin compressing apparatus and a per-pin expanding apparatus wherein a test pattern is handled in the state that the test pattern is divided into a plurality of data each corresponding to a pin. The per-pin architecture has advantages such as (a) since the test data each corresponding to each pin can be down loaded into an internal memory of an automatic test system at the same time, the down loading time of a compressed test pattern can be reduced, (b) since management of the compressing apparatus and the expanding apparatus is simplified, the automatic test system can flexibly cope with increase of the number of pins of an IC.

[0008] A conventional automatic IC testing system (hereinafter referred to as ATE) holds therein a test pattern for testing an IC in a pattern generator. Therefore, in the ATE, a very long time is required for down loading a test pattern from a disk drive storing the test pattern therein to a pattern generator of the ATE. For example, approximately one hour is required, in case of a down loading via a network, for down loading a test pattern having approximately one (1) G bytes (giga bytes). Therefore, it is a serious problem to be solved to decrease the down loading time and to improve an availability factor of the ATE.

DISCLOSURE OF THE INVENTION

[0009] It is an object of the present invention to provide a teat pattern compressing method wherein a data pattern having portions each of which has a different structure or a different statistical characteristic is divided into a plurality of blocks each corresponding to one of those data characteristics and data to be compressed can efficiently be compressed by applying optimum compressing methods to the respective blocks.

[0010] It is another object of the present invention to provide a data pattern expanding method wherein a compressed data divided into a plurality of blocks can be reconstructed to the original data pattern from the respective blocks without any information loss.

[0011] It is still another object of the present invention to provide a test pattern compressing method and a test pattern compressing apparatus wherein a test pattern is divided into test sequences each corresponding to each pin of a semiconductor integrated circuit and the test pattern to be compressed can efficiently be compressed by measuring the structure or the statistical characteristic of the data to apply an optimum compressing method to each test sequence.

[0012] It is still another object of the present invention to provide a test pattern expanding method and a test pattern expanding apparatus wherein a compressed data for each pin of a semiconductor integrated circuit generated by the above compressing method or compressing apparatus can be expanded to the original test pattern data in real time basis for each pin without any information loss.

[0013] It is yet still another object of the present invention to provide a test pattern compressing method and a test pattern expanding method wherein a data pattern divided into a plurality of blocks is compressed by applying an optimum compressing method to each of the blocks and the compressed data divided into those plurality of blocks can be reconstructed to the original data pattern from the respective blocks without any information loss.

[0014] It is yet still another object of the present invention to provide an automatic test system of a semiconductor integrated circuit wherein a down loading time of a test pattern is reduced using the test pattern compressing apparatus and/or the test pattern expanding apparatus.

[0015] The data pattern compressing method according to the present invention is characterized in that in a compressing method wherein an input data having a redundancy, for example, a test pattern data, is compressed in each pin basis of a semiconductor integrated circuit, the data pattern compressing method comprises the steps of: dividing each input data into a plurality of data portions to distribute each portion to one of blocks in accordance with the structure or the statistical characteristic of the data; and applying an appropriate compressing method to each of the blocks; whereby a proper compression can be obtained for all the data.

[0016] The data pattern expanding method according to the present invention, upon expanding the compressed data generated by the data pattern compressing method, comprises the steps of: dividing the compressed data into a plurality of blocks in accordance with the structure of the compressed data; and applying an appropriate expanding method to each of the blocks; whereby the compressed data can be reconstructed to the original data pattern without any information loss.

[0017] In a data pattern compressing method of a preferred embodiment, the dividing step is characterized in that the dividing step includes the steps of: calculating a threshold value of the number of data changes for dividing the input data into blocks; counting the number of data changes of the input data; comparing the actual number of data changes with the threshold value; and distributing a divided data portion to one of the plurality of blocks in accordance with the comparison result; whereby the input data pattern can be divided into proper blocks.

[0018] In the data pattern compressing method of another preferred embodiment, the dividing step is characterized in that the dividing step includes the step of: calculating a threshold value of entropy for dividing the input data into blocks; measuring an appearing probability of each symbol in the input data to calculate a data entropy from the appearing probabilities of symbols; comparing an actual data entropy with the threshold value; and distributing a divided data portion to one of the plurality of blocks in accordance with the comparison result; whereby the input data pattern can be divided into proper blocks.

[0019] In the data pattern compressing method of still preferred another embodiment, the dividing step is characterized in that the dividing step is a step wherein the number of data changes is counted for the input data pattern and the input data is divided into blocks in accordance with the number of data changes; a run length compressing method is applied, in the step of applying the appropriate compressing method, to a block having smaller number of data changes than the threshold value; and respective proper compressing methods are applied to the other blocks; whereby a proper compression can be obtained for all the data.

[0020] In a data pattern expanding method of a preferred embodiment, the dividing step for expanding the compressed data is characterized in that the compressed data is divided into data compressed by the run length compressing method and data compressed by the other plurality of compressing methods and the divided data compressed by the run length compressing method are expanded by a run length expanding method in the step of applying the appropriate expanding method; whereby the compressed data is reconstructed to the original data pattern without any information loss.

[0021] In the data pattern expanding method of still preferred another embodiment, the data pattern expanding method is characterized in that there is included therein a step for performing Burrows Wheeler transform (hereinafter referred to as BW transform) at least once for the block divided in the dividing step and having high periodicity data; the run length compressing method is applied, in the compressing step, to the data transformed by BW transform; and respective appropriate compressing methods are applied to the other blocks; whereby a proper compression is obtained for all the data.

[0022] In the data pattern expanding method of preferred another embodiment, the dividing step for expanding a compressed data is characterized in that the compressed data is divided into data compressed by the run length compressing method after the compressed data is transformed by BW transform and data compressed by the other compressing methods; the portion compressed by the run length compressing method after BW transform is applied is expanded by the run length expanding method in the step of applying an appropriate expanding method and thereafter, inverse BW transform is applied to the portion compressed by the run length compressing method by the number of application times of the BW transform; whereby the compressed data is reconstructed to the original data pattern without any information loss.

[0023] In the data pattern compressing method of still preferred another embodiment, the data pattern compressing method is characterized in that a Huffman compressing method is applied, in the step for applying an appropriate compressing method, to the block having an entropy smaller than a threshold value among the blocks divided in the dividing step and respective appropriate compressing methods are applied to the other blocks: whereby a proper compression can be obtained for all the data.

[0024] In the data pattern expanding method of still preferred another embodiment, the data pattern expanding method is characterized in that a compressed data is divided into data compressed by the Huffman compressing method and data compressed by the other compressing methods; and a Huffman expanding method is applied, in the step of applying an appropriate expanding method, to the data compressed by the Huffman compressing method; whereby the data can be reconstructed to the original data pattern without any information loss.

[0025] In the data pattern compressing method of still preferred another embodiment, the data pattern compressing method is characterized in that an LZ compressing method is applied, in the step of applying an appropriate compressing method, to the block having a small entropy among the divided blocks; and respective optimum compressing methods are applied to the other blocks; whereby a proper compression can be obtained for all the data.

[0026] In the data pattern expanding method of still preferred another embodiment, the data pattern expanding method is characterized in that a compressed data is divided into data compressed by the LZ compressing method and data compressed by the other compressing methods; and an LZ expanding method is applied, in the step of applying an appropriate expanding method, to the data compressed by the LZ compressing method; whereby the compressed data can be reconstructed to the original data pattern without any information loss.

[0027] In the data pattern compressing method of still preferred another embodiment, the data pattern compressing method is characterized in that an arithmetic coding compressing method is applied, in the step of applying an appropriate compressing method, to the block having a small entropy among the divided blocks; and respective appropriate compressing methods are applied to the other blocks; whereby an optimum compression can be obtained for all the data.

[0028] In the data pattern expanding method of still preferred another embodiment, the data pattern expanding method is characterized in that the compressed data is divided, in the dividing step, into data compressed by an arithmetic coding compressing method and data compressed by the other compressing methods; and an arithmetic coding expanding method is applied, in the step of applying an appropriate expanding method, to the data compressed by the arithmetic coding compressing method; whereby the compressed data can be reconstructed to the original data pattern without any information loss.

[0029] In the data pattern compressing method of still preferred another embodiment, the dividing step includes the steps of: inputting environmental parameters for a proper block division; and calculating a threshold value of the number of data changes for the block division using the inputted environmental parameters; whereby an optimum block division can be performed.

[0030] In the data pattern compressing method of still preferred another embodiment, the dividing step includes the steps of inputting environmental parameters for an appropriate block division, and using the inputted environmental parameters to calculate a threshold value of an entropy for a block division.

[0031] In the data pattern compressing method of still preferred another embodiment, the data pattern compressing method includes the steps of calculating a threshold value of the number of data changes for a data division, and using the calculated threshold value to estimate a compression rate of the inputted data and to optimize the threshold value so that the compression rate is maximized, whereby an optimum block division can be made possible using the optimized threshold value in the dividing step.

[0032] In the data pattern compressing method of still preferred embodiment, the data pattern compressing method is characterized in that the data pattern compressing method comprises the steps of calculating a threshold value of entropy for a data division, and using the calculated threshold value to estimate a compression rate of the inputted data and to optimize the threshold value so that the compression rate is maximized, and in that the optimized threshold value is used in the dividing step, whereby an optimum block division can be made possible.

[0033] A test pattern compressing apparatus according to the present invention is characterized in that the test pattern compressing apparatus comprises dividing means for dividing an inputted test pattern into test sequences each corresponding to a pin of an integrated circuit under test, and test sequence compressing means for selecting an appropriate compressing method in accordance with a data structure of each test sequence to compress the test sequence for each pin, whereby the maximum compression rate for all the data can be obtained. In this test pattern compressing apparatus, a test pattern is inputted thereto and data each being compressed in a pin basis can be collectively outputted or can be outputted in the sequence of compression.

[0034] In the test pattern compressing method of further preferred another embodiment, the test pattern compressing method comprises the steps of dividing an inputted test pattern into test sequences each corresponding to a pin of an integrated circuit under test, and selecting an appropriate compressing method in accordance with a data structure of each test sequence to compress the test sequence for each pin.

[0035] In the test pattern compressing apparatus of further preferred another embodiment, the test sequence compressing means is provided for each pin, and those test sequence compressing means operate in parallel to output compressed data in parallel.

[0036] In the test pattern compressing method of still preferred another embodiment, the compressing steps are performed in parallel for respective test sequences divided in pin basis. A high speed and highly efficient compression can be obtained.

[0037] The test pattern expanding apparatus according to the present invention comprises compressed data dividing means for dividing an inputted compressed data into compressed data each corresponding to each pin of an integrated circuit under test, and test sequence expanding means for selecting an appropriate expanding method in accordance with a flag indicating a compressing method for each divided compressed data and for applying the selected expanding method thereto.

[0038] In the test pattern expanding method of still preferred another embodiment, the test pattern expanding method comprises the steps of dividing an inputted compressed data into compressed data each corresponding to each pin of an integrated circuit under test, and selecting an appropriate expanding method in accordance with a flag indicating a compressing method for each divided compressed data and applying the selected expanding method thereto.

[0039] The test pattern expanding apparatus according to the present invention is characterized in that the test pattern expanding apparatus comprises the test sequence expanding means each being for each pin and those test sequence expanding means operate in parallel.

[0040] In the test pattern expanding method of further preferred another embodiment, the expanding steps are performed in parallel in respective pin basis.

[0041] The test sequence compressing means in the test pattern compressing apparatus according to the present invention comprises compressing method determining means for determining a proper compressing method adapted for an inputted test sequence, run length compressing means for compressing a test sequence using a run length compressing method, and BWT run length compressing means for applying a run length compression after applying BW transform at least once to the test sequence.

[0042] This test pattern compressing apparatus also includes, as compressing means, means for using an LZ compressing method, a Huffman compressing method, an arithmetic coding compressing method, and the like as compressing means of further other compressing methods.

[0043] The test pattern compressing method according to the present invention comprises, in the test sequence compressing step, the steps of determining a proper compressing method to be applied to an inputted test sequence, and applying the run length compressing method or applying the run length compressing method after applying BW transform one or more times (herein after referred to as a BWT run length compressing method), or applying any one of the other compressing methods (an LZ compressing method, a Huffman compressing method, an arithmetic coding compressing method and the like) to a test sequence to perform a compression of the test sequence.

[0044] The test sequence expanding apparatus in the test pattern expanding apparatus according to the present invention comprises compressing method determining means for determining whether a compressed data is a data compressed by the run length compressing method, a data compressed by the BWT run length compressing method or a data compressed by one of the other compressing methods such as the LZ compressing method, run length expanding means for performing an expansion of the compressed data using the run length expanding method, repetitive inverse BW transform means for applying inverse BW transform at least once to a data expanded by the run length expanding means to perform a data transform, and other expanding means for performing expansions of the compressed data using expanding methods corresponding to the other compressing methods.

[0045] The test pattern expanding method according to the present invention comprises the steps of determining, in the step of expanding the test sequence, whether a compressed data is a data compressed by the run length compressing method, a data compressed by the BWT run length compressing method or a data compressed by one of the other compressing methods such as the LZ compressing method, and selecting a proper expanding method corresponding to the compressing method for the compressed data to apply the selected expanding method to the compressed data.

[0046] The compressing method determining means in the test pattern compressing apparatus according to the present invention comprises BW transform optimizing means for calculating the proper number of times which is appropriate for applying BW transform to a test sequence, threshold value storage means for storing a threshold value of compression rate, compression rate comparator means for comparing the maximum value of compression rate obtained by the BW transform optimizing means with the threshold value stored in the threshold value storage means, and compressing method selecting means for selecting, in accordance with a comparison result obtained by the compression rate comparator means and the optimum number of times of applying BW transform obtained by the BW transform optimizing means, as to whether the test sequence should be compressed by the run length compressing method, by the BWT run length compressing method, or one of the other compressing methods such as LZ compressing method.

[0047] The BW transform optimizing means in the compressing method determining means comprises the steps of applying BW transform to a test sequence, measuring the number of data changes of a transformed data by applying BW transform, calculating a compression rate from the number of data changes, and obtaining the maximum value of compression rate and the number of application times of BW transform at which the compression rate is maximized.

[0048] A semiconductor integrated circuit automatic test system using the test pattern compressing apparatus and the test pattern expanding apparatus according to the present invention has a large capacity storage device (for example, a disk drive) for storing compressed test pattern data, a pattern generator for storing test pattern data and for generating test patterns, and test pattern expanding means for completely reconstructing the compressed test pattern data to the original test pattern without any information loss, whereby a test pattern down loading time can be reduced.

[0049] The semiconductor integrated circuit automatic test system comprises a large capacity storage device (for example, a disk drive) for storing compressed test pattern data, a pattern generator for storing the test pattern data and generating test patterns, and a plurality of test sequence expanding means each for completely reconstructing a test sequence data compressed in each pin basis to an original test pattern in parallel for respective pins without any information loss.

BRIEF DESCRIPTION OF THE DRAWINGS

[0050]

Fig. 1 is a diagram showing an example of test patterns used in a first embodiment and a second embodiment of the present invention;
Fig. 2 is a block diagram showing an example of a functional construction of a test pattern compressing apparatus used in the first embodiment of the present invention;
Fig. 3 is a flow chart for explaining a test pattern compressing method of the first embodiment of the present invention;
Fig. 4 is a block diagram showing an example of a functional construction of a test pattern expanding apparatus used in the first embodiment of the present invention;
Fig. 5 is a flow chart for explaining a test pattern expanding method of the first embodiment of the present invention;
Fig. 6 is a block diagram showing an example of a functional construction of test pattern dividing means in the test pattern compressing apparatus shown in Fig. 2;
Fig. 7 is a flow chart for explaining a processing procedure of the test pattern dividing means shown in Fig. 6;
Fig. 8 is a diagram showing each example of test patterns, a threshold value of the number of data changes, the number of data changes for each pin, and blocks of division result;
Fig. 9 is a block diagram showing an example of another functional construction of the test pattern dividing means in the test pattern compressing apparatus shown in Fig. 2;
Fig. 10 is a flow chart for explaining a processing procedure of the test pattern dividing means shown in Fig. 9;
Fig. 11 is a diagram showing each example of test patterns, a threshold value of entropy of input data, and blocks of division result in the processing procedure in Fig. 10;
Fig. 12 is a block diagram showing an example of a specific functional construction of the test pattern compressing apparatus;
Fig. 13 is a flow chart for explaining a processing procedure of a run length compressing method;
Fig. 14 is a block diagram showing an example of a functional construction of a test pattern expanding apparatus corresponding to the test pattern compressing apparatus shown in Fig. 12;
Fig. 15 is a flow chart for explaining a processing procedure of a run length expanding method;

Fig. 16 is a block diagram showing an example of another specific functional construction of the test pattern compressing apparatus;

Fig. 17 is a flow chart for explaining a processing procedure of a BW transform method;

Fig. 18 is a block diagram showing an example of a functional construction of a test pattern expanding apparatus corresponding to the test pattern compressing apparatus shown in Fig. 16;

Fig. 19 is a flow chart for explaining a processing procedure of an inverse BW transform method;

Fig. 20 is a block diagram showing an example of further another specific functional construction of the test pattern compressing apparatus;

Fig. 21 is a flow chart for explaining a processing procedure of a Huffman compressing method;

Fig. 22 is a block diagram showing an example of a functional construction of a test pattern expanding apparatus corresponding to the test pattern compressing apparatus shown in Fig. 20;

Fig. 23 is a flow chart for explaining a processing procedure of a Huffman expanding method;

Fig. 24 is a block diagram showing an example of still another specific functional construction of the test pattern compressing apparatus;

Fig. 25 is a flow chart for explaining a processing procedure of an LZ compressing method;

Fig. 26 is a block diagram showing an example of a functional construction of a test pattern expanding apparatus corresponding to the test pattern compressing apparatus shown in Fig. 24;

Fig. 27 is a flow chart for explaining a processing procedure of an LZ expanding method;

Fig. 28 is a block diagram showing an example of still another specific functional construction of the test pattern compressing apparatus;

Fig. 29 is a flow chart for explaining a processing procedure of an arithmetic coding compressing method;

Fig. 30 is a block diagram showing an example of a functional construction of a test pattern expanding apparatus corresponding to the test pattern compressing apparatus shown in Fig. 28;

Fig. 31 is a flow chart for explaining a processing procedure of an arithmetic coding expanding method;

Fig. 32 is a block diagram showing an example of further another specific functional construction of the test pattern dividing means in the test pattern compressing apparatus shown in Fig. 2;

Fig. 33 is a flow chart for explaining a processing procedure of the test pattern dividing means shown in Fig. 32;

Fig. 34 is a block diagram showing an example of still another specific functional construction of the test pattern dividing means in the test pattern compressing apparatus shown in Fig. 2;

Fig. 35 is a flow chart for explaining a processing procedure of the test pattern dividing means shown in Fig. 34;

Fig. 36 is a block diagram showing an example of still another specific functional construction of the test pattern dividing means in the test pattern compressing apparatus shown in Fig. 2;

Fig. 37 is a flow chart for explaining a processing procedure of the test pattern dividing means shown in Fig. 36;

Fig. 38 is a block diagram showing an example of still another specific functional construction of the test pattern dividing means in the test pattern compressing apparatus shown in Fig. 2;

Fig. 39 is a flow chart for explaining a processing procedure of the test pattern dividing means shown in Fig. 38;

Fig. 40 is a block diagram showing an example of still another specific functional construction of the test pattern compressing apparatus;

Fig. 41 is a flow chart for explaining a processing procedure of the test pattern compressing apparatus shown in Fig. 40;

Fig. 42 is a block diagram showing an example of a functional construction of a test pattern expanding apparatus corresponding to the test pattern compressing apparatus shown in Fig. 40;

Fig. 43 is a flow chart for explaining a processing procedure of the test pattern expanding apparatus shown in Fig. 42;

Fig. 44 is a block diagram showing an example of a specific functional construction of the test pattern dividing means in the test pattern compressing apparatus shown in Fig. 40;

Fig. 45 is a flow chart for explaining a processing procedure of the test pattern dividing means shown in Fig. 44;

Fig. 46 is a block diagram showing an example of another specific functional construction of the test pattern dividing means in the test pattern compressing apparatus shown in Fig. 40;

Fig. 47 is a flow chart for explaining a processing procedure of the test pattern dividing means shown in Fig. 46;

Fig. 48 is a block diagram showing an example of further another functional construction of the test pattern compressing apparatus;

Fig. 49 is a flow chart for explaining a processing procedure of the test pattern compressing apparatus shown in Fig. 48;

Fig. 50 is a block diagram showing an example of a functional construction of a repetitive BW transform means in the test pattern compressing apparatus shown in Fig. 48;

Fig. 51 is a flow chart for explaining a processing procedure of the repetitive BW transform means shown in Fig. 50;

Fig. 52 is a flow chart for explaining a processing procedure of the BW transform means in the test pattern compressing apparatus shown in Fig. 48;

Fig. 53 is a diagram showing an example of a circulating character string used in BW transform;

Fig. 54 is a diagram showing a sorting example of the circulating character strings used in BW transform;

Fig. 55 is a diagram showing an example of an output data in BW transform;

Fig. 56 is a block diagram showing an example of further another functional construction of the test pattern expanding apparatus;

Fig. 57 is a flow chart for explaining a processing procedure of the test pattern expanding apparatus shown in Fig. 56;

Fig. 58 is a block diagram showing an example of a functional construction of the repetitive inverse BW transform means in the test pattern expanding apparatus shown in Fig. 56;

Fig. 59 is a flow chart for explaining a processing procedure of the repetitive inverse BW transform means shown in Fig. 58;

Fig. 60 is a flow chart for explaining a processing procedure of the repetitive inverse BW transform means in the test pattern expanding apparatus shown in Fig. 56;

Fig. 61 is a diagram showing an example of correlations between a character string L and a character string F in inverse BW transform;

Fig. 62 is a diagram showing a reconstruction example of a character string S in inverse BW transform;

Fig. 63 is a block diagram showing an example of a functional construction of compressing method determining means in the test pattern compressing apparatus shown in Fig. 48;

Fig. 64 is a flow chart for explaining a processing procedure of the compressing method determining means shown in Fig. 63;

Fig. 65 is a block diagram showing an example of a functional construction of the test pattern compressing apparatus having one input and one output in a second embodiment of the present invention;

Fig. 66 is a flow chart for explaining the test pattern compressing method of the second embodiment of the present invention;

Fig. 67 is a block diagram showing a variation example of a functional construction of the test pattern compressing apparatus having one input and one output in the second embodiment of the present invention;

Fig. 68 is a flow chart for explaining a variation example of the test pattern compressing method of the second embodiment of the present invention;

Fig. 69 is a block diagram showing an example of a functional construction of the test pattern compressing apparatus having one input and multiple outputs of the second embodiment of the present invention;

Fig. 70 is a flow chart for explaining the test pattern compressing method of the second embodiment of the present invention;

Fig. 71 is a block diagram showing an example of a functional construction of the test pattern expanding apparatus having one input and one output of the second embodiment of the present invention;

Fig. 72 is a flow chart for explaining the test pattern expanding method of the second embodiment of the present invention;

Fig. 73 is a block diagram showing a variation example of a functional construction of the test pattern expanding apparatus having one input and one output of the second embodiment of the present invention;

Fig. 74 is a flow chart for explaining a variation example of the test pattern expanding method of the second embodiment of the present invention;

Fig. 75 is a block diagram showing an example of a functional construction of the test pattern expanding apparatus having one input and multiple outputs of the second embodiment of the present invention;

Fig. 76 is a flow chart for explaining the test pattern expanding method of the second embodiment of the present invention;

Fig. 77 is a block diagram showing an example of a functional construction of the test pattern expanding apparatus having multiple inputs and one output of the second embodiment of the present invention;

Fig. 78 is a flow chart for explaining the test pattern expanding method of the second embodiment of the present invention;

Fig. 79 is a block diagram showing a variation example of a functional construction of the test pattern expanding apparatus having multiple inputs and one output of the second embodiment of the present invention;

Fig. 80 is a flow chart for explaining a variation example of the test pattern expanding method of the second embodiment of the present invention;

Fig. 81 is a block diagram showing an example of a functional construction of the test sequence compressing means in the test pattern compressing apparatus shown in Figs. 65, 67 and 69;

Fig. 82 is a flow chart for explaining a processing procedure of the test sequence compressing means shown in Fig. 81;

Fig. 83 is a block diagram showing an example of a functional construction of the repetitive BW transform means in the test sequence compressing means shown in Fig. 81;

Fig. 84 is a flow chart for explaining a processing procedure of the repetitive BW transform means shown in Fig. 83;

Fig. 85 is a flow chart for explaining a processing procedure of the run length compressing means in the test sequence compressing means shown in Fig. 81;

Fig. 86 is a flow chart for explaining a functional construction of the BW transform means in the test sequence compressing means shown in Fig. 81;

Fig. 87 is a diagram showing an example of a circulating character string used in BW transform;

Fig. 88 is a diagram showing a sorting example of circulating character strings used in BW transform;

Fig. 89 is a diagram showing an example of an output data in BW transform;

Fig. 90 is a flow chart for explaining a processing procedure of the LZ compressing means in the test sequence compressing means shown in Fig. 81;

Fig. 91 is a block diagram showing an example of a functional construction of the test sequence expanding means in the test pattern expanding apparatus shown in Figs. 71, 73, 75 and 77;

Fig. 92 is a flow chart for explaining a processing procedure of the test sequence expanding means shown in Fig. 91;

Fig. 93 is a block diagram showing an example of a functional construction of the repetitive inverse BW transform means in the test sequence expanding means shown in Fig. 91;

Fig. 94 is a flow chart for explaining a processing procedure of the repetitive inverse BW transform means shown in Fig. 93;

Fig. 95 is a flow chart for explaining a processing procedure of the run length expanding means in the test sequence expanding means shown in Fig. 91;

Fig. 96 is a flow chart for explaining a processing procedure of the repetitive inverse BW transform means in the test sequence expanding means shown in Fig. 91;

Fig. 97 is a diagram showing an example of correlations between a character string L and a character string F in inverse BW transform;

Fig. 98 is a diagram showing a reconstruction example of a character string S in inverse BW transform;

Fig. 99 is a flow chart for explaining a processing procedure of the LZ expanding means in the test sequence expanding means shown in Fig. 91;

Fig. 100 is a block diagram showing an example of a functional construction of the compressing method determining means in the test sequence compressing means shown in Fig. 81;

Fig. 101 is a flow chart for explaining a processing procedure of the compressing method determining means shown in Fig. 100;

Fig. 102 is a block diagram showing an example of a functional construction of the BWT run length compressing means of the second embodiment of the present invention;

Fig. 103 is a flow chart for explaining a processing procedure of the BWT run length compressing means shown in Fig. 102;

Fig. 104 is a block diagram showing an example of a functional construction of the BWT run length expanding means of the second embodiment of the present invention;

Fig. 105 is a flow chart for explaining a processing procedure of the BWT run length expanding means shown in Fig. 104;

Fig. 106 is a block diagram showing a construction of the automatic test system to which the present invention is applied for testing a semiconductor integrated circuit;

Fig. 107 is a block diagram showing another construction of the automatic test system to which the present invention is applied for testing a semiconductor integrated circuit;

Fig. 108 is a block diagram showing a construction of a measurement system used in the automatic test system shown in Figs. 106 and 107;

Fig. 109 is a block diagram showing further another construction of the automatic test system to which the present invention is applied for testing a semiconductor integrated circuit;

Fig. 110 is a block diagram showing still another construction of the automatic test system to which the present invention is applied for testing a semiconductor integrated circuit;

Fig. 111 is a block diagram showing a construction of a measurement system used in the automatic test system shown in Figs. 109 and 110;

Fig. 112 is a block diagram showing still another construction of the automatic test system to which the present invention is applied for testing a semiconductor integrated circuit;

Fig. 113 is a block diagram showing still another construction of the automatic test system to which the present invention is applied for testing a semiconductor integrated circuit;

Fig. 114 is a graph showing effects to compression rates in the test pattern compressing apparatus according to the present invention;

Fig. 115 is a graph showing effects to expanding speeds in the test pattern expanding apparatus according to the present invention; and

Fig. 116 is a graph showing effects to compression rates in the BWT run length compressing means according to the present invention.

BEST MODES FOR CARRYING OUT THE INVENTION

[0051] Embodiments of the present invention will be explained in detail below with reference to the appended drawings. Further, in order to simplify the explanation below, there will be explained examples in which the present invention is applied to the compression of a test pattern to be applied to an IC and to the expansion of the compressed test pattern to be applied to an IC for testing. However, it is needless to say that the present invention can be applied to the compression and the expansion of a data pattern other than a test pattern to be applied to an IC.

[0052] A test pattern for testing a complex IC is usually generated in order to test certain specific modules within an IC. Therefore, there is a characteristic that only test patterns for specific pins of an IC corresponding to the specific modules frequently change and test patterns for most of the other pins change few.

[0053] Consequently, in a first aspect of the present invention, a test pattern is divided, by paying attention to the structure or the statistical characteristic such as the number of data changes and the like of an input test pattern, into blocks each having related characteristics. Then an appropriate compressing method is applied to each block so that an efficient compression rate can be obtained.

[0054] Fig. 1 shows an example of a test pattern to be applied to an IC. In Fig. 1, lateral direction (left to right direction) is a pin array direction of an IC and longitudinal direction (top to bottom direction) is a sequence direction of test patterns. That is, in the example of Fig. 1, pattern 1 is first applied to pin 1 through pin 6 of an IC. Then test patterns are applied to the IC in the sequence of pattern 2, pattern 3, • • • . In addition, a test pattern is represented by three symbols 1, 0 and X wherein 1 indicates a high level signal, 0 indicates a low level signal and X indicates a "don't care" signal. That is, X may be either a high level signal or a low level signal and therefore indicates a signal which does not influence a test of an IC.

[0055] When a run length compressing method is used for a block division, a threshold value of the number of data changes of a test pattern is first calculated by estimating a compressed data size. Next, the number of data changes of a test pattern for each pin is counted. This actual number of data changes is compared with the threshold value. The test pattern is divided into a block having the number of data changes less than the threshold value and a block having the number of data changes greater than the threshold value. The run length compressing method is applied to the block having the number of data changes less than the threshold value and one of the other appropriate compressing methods is applied to the other blocks. Thus, an efficient compression can be performed.

[0056] For an estimating calculation of a threshold value ϕ_{max} of the number of data changes in the run length compressing method, a minimum value α of an allowable compression rate is given and a maximum value ϕ_{max} satisfying the following formula is obtained.

$$\lceil \mathrm{Log}_2 |A| \rceil \cdot M/\alpha \geq \phi_{max} \lceil \log_2 \{(|A|-1) \cdot M\} \rceil + \lceil \log_2 (\phi_{Max}+2) \rceil \tag{1}$$

In this formula, M is a sequence length of a test pattern in calculating the number of data changes and $|A|$ is size of a set of symbols appearing in a test pattern.

[0057] For example, in the case of a test pattern having 100 pins and 10000 sequences consisting of three signal kinds of "0" low level signal, "1" high level signal and "X" Don't care signal, M is M=10000 and $|A|$ is $|A|=3$. In addition, the symbol $\lceil x \rceil$ means a minimum integer equal to or greater than x.

[0058] By comparing the threshold value obtained as described above with the actual number of data changes, a plurality of blocks each having different number of data changes can be obtained.

[0059] Further, when an attention is paid to an entropy of an input data, an appearing probability p_i ($i=1, \cdots, N$) of a symbol appearing in data is measured and an entropy H of the input data is obtained using the following formula.

$$H=-\Sigma_{i=1}^{N} P_i \log_2 p_i \tag{2}$$

[0060] Next, a threshold value of entropy is calculated by estimating a compression rate of the compressed data. For example, when Huffman's algorithm is used for a threshold value calculation, a threshold value of entropy H_{max} is obtained, by giving a minimum allowable compression rate α, by the following formula.

$$H_{max} \leq \lceil \log_2 |A| \rceil / \alpha \tag{3}$$

[0061] By comparing this threshold value with an actual entropy, a test pattern is divided into blocks each having an entropy less than the threshold value and blocks each having an entropy larger than the threshold value. A Huffman compressing method or an arithmetic coding compressing method is applied to a block having an entropy less than the threshold value and one of the other appropriate compressing methods is applied to the other block. Thus, an efficient compression can be obtained.

[0062] In a second aspect of the present invention, a test pattern is divided into test sequences each being for each pin, and an optimum compressing method is applied, by paying attention to the structure or statistical characteristic of an input test pattern data such as the number of data pattern changes, to each of the test sequences. Thus, an efficient compression rate can be obtained.

[0063] Further, in the test pattern shown in Fig. 1, a row of the test pattern is referred to as a test vector and a column of the test pattern is referred to as a test sequence for a certain pin.

[0064] When an optimum compressing method to be applied to a test sequence is determined, a test sequence is transformed by applying BW transform a predetermined number of times n=N to measure the number of data changes ϕ_i (i=0, 1, • • •, N) for each of 0th BW transform through Nth BW transform. BW transform is described in a paper by Burrows M. and Wheeler D. entitled "A Block-sorting Lossless, Data Compression Algorithm, SRC Research Report 124" Digital Systems Research Center, Palo Alto, CA, May, 1994.

[0065] In this case, the number of data changes represents the number of times that two adjacent data change in a data string like 0 to 1 or 1 to X Then, a compression rate, when the data transformed by BW transform is compressed by a run length compressing method, is calculated from the number of data changes ϕ_i, and the maximum value of the compression rate R_{max} and the number of times for applying BW transform m by which the compression rate is maximized are acquired.

[0066] Then, the acquired maximum value R_{max} of the compression rate is compared with a threshold value of the compression rate R_{th} which is experimentally set. If the maximum value of the compression rate is equal to or grater than the threshold value, a test sequence is transformed m times by BW transform and thereafter the run length compressing method is applied to the test sequence for compression. In this case, if the number of application times of BW transform m is zero (0), BW transform is not applied to the test sequence and only the run length compressing method is applied thereto for compression. If the maximum value of the compression R_{max} is less than the threshold value R_{th}, only the LZ compressing method is applied to the test sequence for compression. In such a way, an efficient compression can be performed by applying an optimum compressing method to the test sequence.

[0067] Further, in the present invention, since a test pattern is divided into test sequences each being for each pin, a compression of each test sequence is made possible. Therefore, a real time test pattern expansion can be performed by using an expanding apparatus for each pin.

[0068] In order to calculate a compression rate Ri when a data transformed i times by BW transform using the number of data changes is compressed by the run length compressing method, the compression rate Ri is obtained by the following formula using the number of data changes ϕ_i and the number of application times of BW transform.

$$R_i=\log_2 |A| \cdot L/[i \cdot (\log_2 L)+\phi_i \cdot [\log_2 \{(|A|-1) \cdot L)\}] +\log_2 (L-1)+\log_2 |A|] \qquad (4)$$

In this case, L is a length of a test sequence and |A| is a size of a set of symbols appearing in a test pattern.

[0069] For example, in the case of a test pattern having 100 pins and 10000 sequences consisting of three signal kinds of "0" low level signal, "1" high level signal and "X" Don't care signal, L is L=10000 and |A| is |A|=3. In addition, a symbol [x] means a minimum integer greater than x.

[0070] From the compression rate R_i (i=0, 1, • • •, N) obtained by a method as described above, the maximum compression rate R_{max} and the number of application times m of BW transform by which the compression rate is maximized are acquired. By comparing the R_{max} with the threshold value R_{th}, an optimum compressing method can be determined.

[0071] Since, as described above, each test sequence can be compressed by an optimum compressing method, an entire test pattern can be efficiently compressed. Therefore, by compressing data in a test pattern to a smaller data using the compressing method according to the present invention, data transmission time can be reduced. As a result, a reduction of a down loading time of a test pattern can be made possible.

[0072] Examples of applying the block dividing method to a test pattern will be explained in detail by showing specific examples.

[0073] Fig. 2 is a block diagram showing an example of a functional construction of the test pattern compressing apparatus used in a first embodiment of the present invention. This test pattern compressing apparatus 21 comprises test pattern dividing means 22 for dividing a test pattern into a plurality of blocks in accordance with a data structure or a statistical characteristic of an inputted test pattern and a plurality of data compressing means A23, B24, C25, • • • for compressing the divided data.

[0074] Next, the operations for compressing a test pattern using this test pattern compressing apparatus 21 will be explained.

[0075] Fig. 3 is a flow chart for explaining the test pattern compressing method according to a first embodiment of the present invention. The test pattern dividing means 22 in the test pattern compressing apparatus 21 shown in Fig. 2 classifies, in step 31, a test pattern for each pin into one of a plurality of blocks in accordance with a data structure or a statistical characteristic of the data to divide the entire test pattern.

[0076] A data pattern of each divided block is transmitted via a switch S1, in step 32, to one of a plurality of data compressing means A23, B24, C25, • • • each having an optimum compression algorithm for corresponding one of the blocks. Then, in steps 33a, 33b and 33c, the data patterns are compressed by the respective data compressing means. Finally, in step 34, each compressed data is outputted through a switch S2.

[0077] In this case, a flag 1 indicating the compressing method of the data compressing means A23 is affixed to the front portion of the compressed data from the data compressing means A23. Flags 2 and 3 indicating respective compressing methods of the data compressing means A23 and B24 are affixed to the front portions of the compressed data from the data compressing means A23 and B24, respectively. In addition, for example, a pin number indicating that a test pattern corresponds to which one of the pins is affixed to each test pattern.

[0078] Fig. 4 is a block diagram showing an example of a functional construction of the test pattern expanding apparatus used in the first embodiment of the present invention. The test pattern expanding apparatus 41 comprises compressed data dividing means 42 for classifying compressed data to be expanded into data groups each having been compressed by a different compressing method, and a plurality of data expanding means A43, B44, C45, • • • for expanding the classified compressed data to their original test patterns without any information loss. In this case, the data expanding means 43, 44, 45, • • • correspond to the compressing means 23, 24, 25, • • • in the test pattern compressing apparatus 21, respectively.

[0079] Next, the operations for expanding the compressed data using this test pattern expanding apparatus 41 will be explained. Fig. 5 is a flow chart for explaining a test pattern expanding method of the first embodiment of the present invention.

[0080] First, in step 51, the compressed data dividing means 42 divides the compressed data to be expanded into data groups each having been compressed by a different compressing method. This division is performed based on, for example, flags 1, 2 and 3 indicating the respective compressing methods. The divided compressed data are transmitted, in step 52, to the respective appropriate data expanding means A43, B44, C45, • • • via a switch S3. That is, a compressed data having a flag 1 is transmitted to the data expanding means 43, and compressed data having flags 2 and 3 are transmitted to the respective data expanding means 44 and 45. The transmitted data are, in step 53, expanded to their original blocks by the respective data expanding means without any information loss. Finally, in step 54, an expanded test pattern for each pin is outputted via a switch S4.

[0081] Fig. 6 is a block diagram showing an example of a specific functional construction of the test pattern dividing means 22 used in the first embodiment of the present invention. This test pattern dividing means 22 comprises parameter inputting means 62 for inputting environmental parameters for a block division, parameter storage means 63 for storing the inputted environmental parameters, threshold value calculating means 64 for calculating a threshold value of the number of data changes for dividing a test pattern into blocks using the environmental parameters stored in the parameter storage means 63, number of data changes counting means 65 for counting the number of data changes of the inputted test pattern, number of data changes comparator means 66 for comparing the threshold value of the number of data changes with the actual number of data changes, and block dividing means 67 for dividing an inputted test pattern into a plurality of blocks in accordance with the comparison results.

[0082] Next, the operations for dividing a test pattern into blocks using this test pattern dividing means 22 will be explained. Fog. 7 is a flow chart for explaining a test pattern dividing method of the first embodiment of the present invention. In addition, Fig. 8 shows an example in which this test pattern dividing method is applied to a test pattern, wherein Fig. 8(a) is an example of an input test pattern to be used in the explanation below. In the test pattern shown in Fig. 8 (a), a lateral direction (left to right direction in the figure) is a pin array direction of an integrated circuit and a longitudinal direction (top to bottom direction in the figure) is a pattern sequence direction.

[0083] First, using the parameter inputting means 62, environmental parameters for calculating a threshold value of block division, i.e., a length M of a test pattern sequence for calculation of the number of data changes, a size $|A|$ of a set of symbols appearing in a test pattern, and a lowest limit value α of allowable compression rate are inputted. Those inputted parameters are stored in the parameter storage means 63.

[0084] Next, using the parameters stored in the parameter storage means 63, in step 72, the threshold value calculating means 64 calculates a threshold value of the number of data changes in a pattern for dividing an inputted test pattern into a plurality of blocks. The formula (1) is used for this calculation. In the example shown in Fig. 8, if $\alpha=1$ is assumed, the threshold value ϕ_{max} is obtained, as shown in Fig. 8 (b), because of $\phi_{max}=2$ from the parameters $M=6$ and $|A|=3$.

[0085] Next, in step 73, the number of data changes counting means 65 counts the number of data changes in a test pattern for each pin of the inputted test pattern. In the example shown in Fig. 8, the number of data changes, for example, in the pattern for pin 1 is zero (0) and the number of data changes in the pattern for pin 2 is 1, i.e. one change from

0 to X. In such a way, as shown in Fig. 8 (c), the number of data changes ϕ can be obtained as ϕ="013205". The threshold value ϕ_{max} calculated by the threshold value calculating means 64 and the number of data changes ϕ obtained by the number of data changes counting means 65 are transmitted to the number of data changes comparator means 66 to be compared with each other. The comparison results are sent to the block dividing means 67.

[0086] Next, in steps 75 and 76, the block dividing means 67 classifies the test pattern data in pin basis into a plurality of blocks in accordance with the corresponding number of data changes, i.e., distributes the test pattern data for each pin into one of the plurality of blocks. In the example shown in Fig. 8, the test pattern data in pin basis are classified into a block in which a test pattern data has the number of data changes equal to or less than the threshold value ϕ_{max} and a block in which a test pattern data has the number of data changes greater than the threshold value ϕ_{max}. Finally, the block dividing means 67 sends the data (patterns) in each block which have been classified (distributed) in step 77 to a plurality of corresponding compressing means.

[0087] Fig. 9 is a block diagram showing an example of another specific functional construction of the test pattern dividing means 22 used in the first embodiment of the present invention. This test pattern dividing means 22 comprises parameter inputting means 92 for inputting environmental parameters for a block division, parameter storage means 93 for storing the inputted environmental parameters, threshold value calculating means 94 for calculating a threshold value of entropy for dividing a test pattern using the environmental parameters stored in the parameter storage means 93, entropy measuring means 95 for measuring an appearing probability of each symbol appearing in a inputted data and for calculating an entropy of the data from the appearing probabilities of the symbols, entropy comparator means 96 for comparing the threshold value of entropy with an entropy of actual data, and block dividing means 97 for dividing an input test pattern in accordance with the comparison result of the comparator means 96.

[0088] Next, the operations for dividing a test pattern into blocks using this test pattern dividing means 22 will be explained. Fig. 10 is a flow chart for explaining a test pattern dividing method of the first embodiment of the present invention. Fig. 11 shows an example in which this block dividing method is applied to a test pattern. Fig. 11(a) is an example of an input test pattern to be used in the explanation below. Here, in the test pattern shown in Fig. 11(a), a lateral direction (left to right direction in the figure) is a pin array direction of an IC and longitudinal direction (top to bottom direction in the figure) is a pattern sequence direction.

[0089] First, in step 101, parameters for calculating a threshold value for a block division are inputted using the parameter inputting means 92 to be stored in the parameter storage means 93. Next, in step 102, the threshold value calculating means 94 calculates a threshold value of entropy for dividing an inputted test pattern into a plurality of blocks. The formula (3) is used for calculation of the threshold value. In the example shown in Fig. 11, assuming that α is $\alpha=2$, the threshold value H_{max} is obtained, as shown in Fig. 8(b), because of $H_{max}=1$ since $|A|$ is $|A|=3$.

[0090] Next, in step 103, symbol appearing probabilities in an inputted test pattern are measured. In addition, in step 104, an entropy of the inputted test pattern is calculated using the formula (2) from the probabilities obtained in step 103. Those steps 103 and 104 are performed in the entropy counting means 95. In the example shown in Fig. 11, the symbol appearing probabilities p for pin 1 are obtained as 0 from the formula (2) because of $p(0)=1$, $p(1)=0$ and $p(X)=0$. Further, for pin 2, $p(0)=1/3$, $p(1)=0$ and $p(X)=2/3$, and hence the entropy is obtained as 0.92.

[0091] Similarly, the entropy H for the respective pins can be calculated to obtain H=[0.0 0.92 1.46 0.92 0.0 1.46]. The threshold value H_{max} calculated by the threshold value calculating means 94 and the entropy H obtained by the entropy counting means 95 are sent to the entropy comparator means 96 to be compared with each other in step 105.

[0092] The comparison result is transmitted to the block dividing means 97. Then in steps 106 and 107, the block dividing means 97 divides each test pattern data for each pin into a plurality of blocks in accordance with the corresponding entropy. In the example shown in Fig. 11, an inputted test pattern is divided into a block in which a test pattern for a pin has, as shown in Fig. 11(d), an entropy equal to or less than the threshold value H_{max} and a block in which a test pattern for a pin has an entropy greater than the threshold value H_{max}. Finally, the block dividing means 97 sends, in step 108, the data in each divided block to corresponding one of the plurality of compressing means.

[0093] Next, variation examples of the test pattern compressing apparatus, the test pattern expanding apparatus and the test pattern dividing means 22 will be explained.

[0094] Fig. 12 is, in the first embodiment of the present invention, a block diagram showing an example of a functional construction of the test pattern compressing apparatus 21 in the case where one of the data compressing means of the test pattern compressing apparatus shown in Fig. 2 is run length compressing means. This test pattern compressing apparatus 21 comprises test pattern dividing means 122 for counting the number of data changes of an inputted test pattern and for performing a block division of an inputted data in accordance with the number of data changes, run length compressing means 123 for applying a run length compressing method to a block having a small number of data changes, i.e., a block having the number of data changes equal to or less than the threshold value, and at least one of the data compressing means 124, 125, • • • for applying another method such as, for example, an LZ compressing method, a Huffman compressing method or the like to the other blocks.

[0095] A flag indicating the compressing method is affixed to the front portion of each data compressed by one of the compressing means 123, 124, • • • .

[0096] Next, the operations for compressing data using the run length compressing means 123 will be explained. Fig. 13 is a flow chart for explaining a processing procedure of the run length compressing method and shows the run length compressing method of a variation example 1.

[0097] First, in step 131, an initial value and a counter value for generating run length codes are set. Next, in step 132, one character is acquired from the input data, and in step 133, the number of characters is counted. In step 134, the acquired character is compared with the initial value. If the character is equal to the initial value, the steps 132 and 133 are repeated. If the character is different from the initial value, the count value up to then is coded, in step 135, as a length that the initial value has continued. In step 136, a check is made to see if there are remaining input data. If there are remaining input data, the steps up to then are repeated. If there is no remaining input data, compressed data are outputted in step 137 and the process ends.

[0098] Fig. 14 is, in the first embodiment of the present invention, a block diagram showing an example of a construction of the test pattern expanding apparatus 41 in the case where one of the data expanding means is the run length expanding means. This test pattern expanding apparatus 41 comprises compressed data dividing means 142 for dividing a compressed data to be expanded into data which have been compressed by the run length compressing method and data which have been compressed by the other plurality of compressing methods using flags each indicating a compressing method, run length expanding means 143 for expanding each of the divided compressed data to the original test pattern without any information loss, and data expanding means 144, 145, • • • according to at least one of the other compressing methods such as, for example, an LZ expanding method, a Huffman expanding method or the like.

[0099] Next, the operations for performing a data expansion using the run length expanding means 143 will be explained. Fig. 15 is a flow chart for explaining a processing procedure of the run length expanding method and shows the run length expanding method of a variation example 1 of the present invention.

[0100] First, in step 151, a compressed data is acquired and in step 152, an initial value and the number of continued characters are acquired from the compressed data. Then, in step 153, the initial value is copied by the number of times equal to the number of continued characters. In step 154, a check is made to see if there are remaining compressed data. If there are remaining compressed data, the steps 151, 152 and 153 are repeated. If there is no remaining compressed data, expanded data are outputted in step 155 and the process ends.

[0101] Fig. 16 is, in the first embodiment of the present invention, a block diagram showing an example of a construction of the test pattern compressing apparatus 21 in the case where one of the data compressing means is the run length compressing means and BW transform means is provided in the front stage of the run length compressing means. This test pattern compressing apparatus 21 comprises test pattern dividing means 162 for dividing an input data into blocks in accordance with a structure or a statistical characteristic of the inputted test pattern data, BW transform means 163 for applying BW transform to a block having a large data periodicity, i.e., a block having the number of data changes greater than the threshold value and having an entropy equal to or less than the threshold value, run length compressing means 164 for applying the run length compressing method to a data to which BW transform has been applied, and at least one of the data compressing means 165 and 166 for applying other data compressing methods to the other blocks.

[0102] Next, the operations for performing a transform using the BW transform means 163 will be explained. Fig. 17 is a flow chart for explaining a processing procedure of a BW transform method and shows the BW transform method of a variation example 2 of the first embodiment of the present invention.

[0103] First, in step 171, a character string S having n characters is acquired. Then in step 172, n circulating character strings (character strings cyclically shifted) of the acquired character string S are generated. Then in step 173, those circulating character strings are sorted to create a matrix M. In steps 174 and 175, a character string L obtained from the last column of the matrix M and a number I of a row having the same character string as the input character string S are acquired from the created matrix M. Finally, in step 176, a check is made to see if there are remaining input data. If there are remaining input data, those steps 171, 172, 173, 174 and 175 are repeated. If there is no remaining input data, compressed data are outputted in step 177 and the process ends.

[0104] Fig. 18 is, in the first embodiment of the present invention, a block diagram showing an example of a construction of the test pattern expanding apparatus in the case where one of the data expanding means is the run length expanding means and the BW transform means is provided at the rear stage of the run length compressing means. This test pattern expanding apparatus 41 comprises compressed data dividing means 182 for dividing a compressed data to be expanded into data which have been compressed by the run length compressing method after BW transform is applied and data which have been compressed by the other plurality of compressing methods, run length expanding means 183 for expanding the divided compressed data to the original test pattern without any information loss, inverse BW transform means 184 for applying inverse BW transform to the data expanded by the run length expanding means, and at least one of the data expanding means 185 and 186 for applying other compressing methods to the other data.

[0105] Next, the operations for performing the inverse transform using the inverse BW transform means will be explained. Fig. 19 is a flow chart for explaining a processing procedure of an inverse BW transform method and shows the inverse BW transform method of the variation example 2 of the first embodiment of the present invention.

[0106] First, in step 191, a character string L having n characters and a number I are acquired. Then in step 192, characters in the acquired character string L are sorted to generate a character string F. In step 193, a matrix T indicating correlations between characters in the character string L and characters in the character string F is created. Then in step 194, the original character string S is reconstructed from the character string L, the number I and the matrix T. Finally in step 195, a check is made to see if there are remaining data. If there are remaining data, those steps 191, 192, 193 and 194 are repeated. If there is no remaining input data, inversely transformed data are outputted in step 196 and the process ends.

[0107] Fig. 20 is, in the first embodiment of the present invention, a block diagram showing an example of a construction of the test pattern compressing apparatus 21 in the case where one of the data compressing means is Huffman compressing means. This test pattern compressing apparatus 21 comprises test pattern dividing means 202 for dividing an input data into blocks in accordance with a statistical characteristic of the inputted test pattern data, Huffman compressing means 203 for applying a Huffman compressing method to a block having a small entropy, i.e., a block having an entropy equal to or less than the threshold value, and at least one of the data compressing means 204 and 205 for compressing data of the other blocks by other methods such as, for example, an LZ compressing method, an arithmetic coding compressing method or the like.

[0108] Next, the operations for performing a data compression using the Huffman compressing means 203 will be explained. Fig. 21 is a flow chart for explaining a processing procedure of the Huffman compressing method and shows the Huffman compressing method of a variation example 3 of the first embodiment of the present invention.

[0109] First, in step 211, an appearing probability of each character appearing in an input data is measured. Then in step 212, a binary tree for creating Huffman codes is created from the acquired appearing probabilities. In step 213, a code is assigned to each character based on the Huffman's binary tree. Then, in step 214, one character is acquired from the input data. In step 215, the acquired character is converted to the assigned code. Finally, in step 216, a check is made to see if there are remaining input data. If there are remaining input data, the steps 214 and 215 are repeated. If there is no remaining input data, compressed data are outputted in step 217 and the process ends,

[0110] Fig. 22 is, in the first embodiment of the present invention, a block diagram showing an example of a construction of the test pattern expanding apparatus 41 in the case where one of the data expanding means is the Huffman expanding means. This test pattern expanding apparatus 41 comprises compressed data dividing means 222 for dividing a compressed data to be expanded into data compressed by the Huffman compressing method and data compressed by the other plurality of compressing methods, Huffman expanding means 223 for expanding the divided compressed data to the original test pattern without any information loss, and at least one of the data expanding means 224 and 225.

[0111] Next, the operations for performing a data expansion using the Huffman expanding means 223 will be explained. Fig. 23 is a flow chart for explaining a processing procedure of a Huffman expanding method and shows the Huffman expanding method of a variation example 3 of the first embodiment of the present invention.

[0112] First, in step 231, the binary tree used in the compression is acquired from the compressed data. Since Huffman's binary tree must be traced in order to obtain an original character from a code, the process moves, in step 232, to a root node of the binary tree. Then in step 233, a binary code is acquired from the compressed data and in step 234, the Huffman's binary tree is traced based on the acquired binary code. After that in step 235, a check is made to see if the current node is a leaf node of the binary tree. If the current node is not a leaf node, the steps 233 and 234 are repeated until the process reaches a leaf node. If the current node is a leaf node, a character assigned to the leaf node is outputted in step 236. Finally in step 237, a check is made to see if there are remaining binary codes. If there are remaining binary codes, those steps 231, 232, 233, 234, 235 and 236 are repeated. If there is no remaining input data, expanded data are outputted in step 238 and the process ends.

[0113] Fig. 24 is, in the first embodiment of the present invention, a block diagram showing an example of a construction of the test pattern compressing apparatus 21 in the case where one of the data compressing means is LZ compressing means. This test pattern compressing apparatus 21 comprises test pattern dividing means 242 for dividing an input data into blocks in accordance with a statistical characteristic of the input test pattern data, LZ compressing means 243 for applying an LZ compressing method to a block having a small entropy, i.e., a block having an entropy equal to or less than the threshold value, and at least one of the other data compressing means 244 and 245 for compressing data of the other blocks by other methods such as, for example, a Huffman compressing method, an arithmetic coding compressing method or the like.

[0114] Next, the operations for performing a data compression using the LZ compressing means 243 will be explained. Fig. 25 is a flow chart for explaining a processing procedure of the LZ compressing method and shows the LZ compressing method of a variation example 4 of the first embodiment of the present invention.

[0115] First, in step 251, a dictionary to be used in the data compression is initialized. Then in step 252, an input data is acquired and in step 253, a matching between the acquired character string and the dictionary is performed. In step 254, a check is made to see if there is a character string that matches better with the dictionary. If there is a possibility that there may be a character string matching better with the dictionary, the matching in step 253 is repeated. If there is no character string that matches better with the dictionary, characters matched with the dictionary and the first character which did not match with the dictionary are coded in step 255. Then the new character string is registered in the dictionary in step 256. Finally in step 257, a check is made to see if there are remaining input data. If there remaining input data, the steps 252, 253, 254, 255 and 256 are repeated. If there is no remaining input data, compressed data are outputted in step 258 and the precess ends.

[0116] Fig. 26 is, in the first embodiment of the present invention, a block diagram showing an example of a construction of the test pattern expanding apparatus 41 in the case where one of the data expanding means is the LZ expanding means. This test pattern expanding apparatus 41 comprises compressed data dividing means 262 for dividing a compressed data to be expanded into data which have been compressed by the LZ compressing method and data which have been compressed by the other plurality of compressing methods, LZ expanding means 263 for expanding the compressed data to the original test pattern without any information loss, and at least one of the other data expanding means 264 and 265 having other expanding methods.

[0117] Next, the operations for performing a data expansion using the LZ expanding means 263 will be explained. Fig. 27 is a flow chart for explaining a processing procedure of an LZ expanding method and shows the LZ expanding method of a variation example 4 of the first embodiment of the present invention.

[0118] First, in step 271, a dictionary to be used in the data expansion is initialized. Then in step 272, a code is acquired from an input compressed data and in step 273, a search in the dictionary is performed using the acquired code. Further in step 274, the code is converted to a character string and then in step 275, the newly generated character string is registered in the dictionary. Finally in step 276, a check is mode to see if there are remaining compressed data. If there are remaining compressed data, the steps 272, 273, 274, 275 are repeated. If there is no remaining compressed data, expanded data are outputted in step 277 and the process ends.

[0119] Fig. 28 is, in the first embodiment of the present invention, a block diagram showing an example of a construction of the test pattern compressing apparatus 21 in the case where one of the data compressing means is arithmetic coding compressing means. This test pattern compressing apparatus 21 comprises test pattern dividing means 282 for dividing an input data into blocks in accordance with a statistical characteristic of the inputted test pattern data, arithmetic coding compressing means 283 for applying an arithmetic coding to a block having a small entropy, i.e., a block having an entropy equal to or less than the threshold value, and at least one of the data compressing means 284 and 285 for compressing data of the other blocks by other methods such as, for example, the LZ compressing method, the Huffman compressing method, or the like.

[0120] Next, the operations for performing a data compression using the arithmetic coding compressing means 283 will be explained. Fig. 29 is a flow chart for explaining a processing procedure of the arithmetic coding compressing method and shows an arithmetic coding compressing method of a variation example 5 of the first embodiment of the present invention.

[0121] First, in step 291, an appearing probability of each character in an input data is measured and then in step 292, a segment on a numerical straight line starting from 0 to 1 is allocated to each character. Then in step 293, a character is acquired from the input data and in step 294, the acquired character is converted to the segment on the numerical straight line. Finally in step 295, a check is made to see if there are remaining input data. If there are remaining input data, the steps 293 and 294 are repeated to further narrow the range of the segment on the numerical straight line. If there is no remaining input data, the lowest limit value of the segment obtained is acquired in step 296. Then in step 297, compressed data are outputted and the process ends.

[0122] Fig. 30 is, in the first embodiment of the present invention, a block diagram showing an example of a construction of the test pattern expanding apparatus 41 in the case where one of the data expanding means is the arithmetic coding expanding means. This test pattern expanding apparatus 41 comprises compressed data dividing means 302 for dividing a compressed data to be expanded into data compressed by the arithmetic coding compressing method and data compressed by the other plurality of compressing methods, arithmetic coding expanding means 303 for expanding each compressed data to the original test pattern without information loss, and at least one of the other data expanding means 304 and 305 using the other expanding methods.

[0123] Next, the operations for performing a data expansion using the arithmetic coding expanding means 303 will be explained. Fig. 31 is a flow chart for explaining a processing procedure of the arithmetic coding expanding method and shows the arithmetic coding expanding method of the variation example 5 of the first embodiment of the present invention.

[0124] First, in step 311, information on an appearing probability of each character is acquired from a compressed data and a segment on a numerical straight line is allocated, quite similarly to the case of data compression, to each character. Then in step 312, a real number value code is acquired from the compressed data and in step 313, a character allocated to a segment in which a real number is included is outputted. Further, in steps 314 and 315, the lowest limit value of the segment is subtracted from the real number value. In addition, the real number value is divided by the segment width. Finally, in step 316, a check is made to see if the real number value is equal to zero. If the real number value is not zero, the steps 313, 314, 315 are repeated until the real number value becomes zero. If the real number value becomes zero, expanded data are outputted in step 317 and the process ends.

[0125] Fig. 32 is a block diagram showing an example of an internal construction of the test pattern dividing means 22 used in the first embodiment of the present invention. This test pattern dividing means 22 comprises threshold value storage means 322 for storing a predetermined threshold value, number of data changes counting means 322 for counting the number of data changes of an input data, number of data changes comparator means 324 for comparing the threshold value with the actual number of data changes, and block dividing means 325 for dividing the input data into a plurality of blocks in accordance with the comparison result.

[0126] Next, the operations for dividing a test pattern into blocks using the test pattern dividing means 22 will be explained. Fig. 33 is a flow chart for explaining a processing procedure of the test pattern dividing means 22 shown in Fig. 32 and shows a test pattern dividing method of the first embodiment of the present invention.

[0127] First, in step 331, the number of data changes counting means 323 counts the number of data changes of a test pattern for each pin of an inputted test pattern. A threshold value stored in the threshold value storage means 322 and the number of data changes obtained by the number of data changes counting means 325 are transmitted to the number of data changes comparator means 324 where, in step 332, the both data are compared with each other. The comparison result is transmitted to the block dividing means 325. Then in steps 333 and 334, the block dividing means 325 divides the test pattern data into a plurality of blocks in accordance with the number of data changes of each test pattern data. Finally, the block dividing means 325 sends the data of each divided block to a corresponding one of the plurality of the data compressing methodS.

[0128] Fig. 34 is a block diagram showing another example of an internal construction of the test pattern dividing means 22 used in the first embodiment of the present invention. This test pattern dividing means 22 comprises threshold value storage means 342 for storing a predetermined threshold value, entropy measuring means 343 for measuring an appearing probability of each of symbol in an input data and for calculating an entropy of the data from the appearing probabilities of the symbols, entropy comparator means 344 for comparing the threshold value with an entropy of an actual data, and block dividing means 345 for dividing the input data into a plurality of blocks in accordance with the comparison results.

[0129] Next, the operations for dividing a test pattern into blocks using this test pattern dividing means 22 will be explained. Fig. 35 is a flow chart for explaining a processing procedure of the test pattern dividing means 22 shown in Fig. 34 and shows a test pattern dividing method of the first embodiment of the present invention.

[0130] First, in step 351, an appearing probability of each symbol in the inputted test pattern is measured. In addition, in step 352, an entropy of the input data is calculated from the probabilities obtained in step 351. The above steps 351 and 352 are performed by the entropy measuring means 343.

[0131] The threshold value stored in the threshold value storage means 342 and the entropy obtained by the entropy measuring means 343 are sent to the entropy comparator means 344 where those data are compared with each other in step 353. The comparison result is sent to the block dividing means 345. Next, in steps 354 and 355, the block dividing means 345 divides the test pattern data into a plurality of blocks in accordance with the respective entropies. Finally, the block dividing means 345 transmits, in step 356, the data of each divided block to corresponding one of the plurality of data compressing means.

[0132] Fig. 36 is a block diagram showing another example of an internal construction of the test pattern dividing means 22 used in the first embodiment of the present invention. This test pattern dividing means 22 comprises threshold value calculating means 362 for calculating a threshold value of the number of data changes for a block division, optimizing means 363 for estimating a compression rate using the threshold value calculated by the threshold value calculating means 362 and for optimizing the threshold value so as to maximize the compression rate, number of data changes counting means 364 for counting the number of data changes, number of data changes comparator means 365 for comparing the optimized threshold value with the actual number of data changes, block dividing means 366 for dividing the test pattern into a plurality of blocks in accordance with the comparison results.

[0133] Next, the operations for dividing a test pattern into blocks using this test pattern dividing means 22 will be explained. Fig. 37 is a flow chart for explaining a processing procedure of the test pattern dividing means 22 shown in Fig. 36 and shows a test pattern dividing method of the first embodiment of the present invention.

[0134] First, in step 371, all the cases that are possible to occur are listed. Then in step 372, the threshold value calculating means 362 selects one of the cases listed in the step 371 to calculate a threshold value of the number of data changes of the pattern for dividing an input test pattern into a plurality of blocks. Then in step 373, the block division optimizing means 363 performs a block division and a data compression using the threshold value obtained in the step 372 and calculates a compression rate.

[0135] Next, in step 374, the compression rate is compared with the previous result. If the compression rate is higher than the previous result, the threshold value is updated to a new value in step 375. If the compression rate is lower than the previous result, the process proceeds to next step 376. In the step 376, a check is mate to see if there are the other cases not verified yet. If there are the other cases not verified yet, the steps 372, 373, 374 and 375 are repeated. If there is no case to be verified, the process moves to step 377. In the step 377, the number of data changes counting means 364 counts the number of data changes of a test pattern for each pin of the input test pattern. The optimized threshold value and the number of data changes obtained by the number of data changes counting means 364 are sent to the number of data changes comparator means 365 where the both data are compared with each other. The comparison result is sent to the block dividing means 366.

[0136] Next, in steps 379 and 3710, the block dividing means 366 divides the test pattern data into a plurality of blocks in accordance with the respective number of data changes. Finally, in step 3711, the block dividing means 366 transmits the data of each divided block to corresponding one of the plurality of data compressing means.

[0137] Fig. 38 is a block diagram showing further another example of an internal construction of the test pattern dividing means 22 used in the first embodiment of the present invention. This test pattern dividing means 22 comprises threshold value calculating means 382 for calculating a threshold value of entropy for a block division, optimizing means 383 for estimating a compression rate using the threshold value calculated by the threshold value calculating means 382 and for optimizing the threshold value so as to maximize the compression rate, entropy measuring means 384 for measuring an appearing probability of each symbol in an input data and for calculating an entropy of the data from those probabilities of the symbols, entropy comparator means 385 for comparing the optimized threshold value with an entropy of the actual data, and block dividing means 386 for dividing the input data into a plurality of blocks in accordance with the comparison results.

[0138] Next, the operations for dividing a test pattern into blocks using this test pattern dividing means 22 will be explained. Fig. 39 is a flow chart for explaining a processing procedure of the test pattern dividing means shown in Fig. 38 and shows a test pattern dividing method of the first embodiment of the present invention.

[0139] First, in step 391, all the cases that are possible to occur are listed. Then in step 392, the threshold value calculating means 382 selects one of the cases listed in the step 391 to calculate a threshold value of entropy for dividing an input test pattern into a plurality of blocks. Then in step 393, the block division optimizing means 383 performs a block division and a data compression using the threshold value obtained in the step 392 and calculates a compression rate. Then in step 394, the compression rate is compared with a previous result. If the compression rate is higher than the previous result, the threshold value is updated to a new value in step 395. If the compression rate is lower than the previous result, the process proceeds to next step 396.

[0140] In the step 396, a check is made to see if there are the other cases not verified yet. If there are the other cases not verified yet, the steps 392, 393, 394 and 395 are repeated. If there is no case to be verified, the process moves to step 397. In the step 397, an appearing probability of each symbol in the inputted test pattern is measured. In addition, in step 398, an entropy of the input data is calculated from the probabilities obtained in the step 397. The above steps 397 and 398 are performed by the entropy measuring means 384. The optimized threshold value and the entropy obtained by the entropy measuring means 384 are sent to the entropy comparator means 385 where those data are compared with each other in step 399. The comparison result is transmitted to the block dividing means 386.

[0141] Next, in steps 3910 and 3911, the block dividing means 386 divides the test pattern data into a plurality of blocks in accordance with the respective entropies. Finally, the block dividing means 386 transmits, in step 399, the data of each divided block to corresponding one of the plurality of data compressing method.

[0142] Fig. 40 is, in the first embodiment of the present invention, a block diagram showing an example of a construction of the test pattern compressing apparatus in the case where the test pattern compressing apparatus has run length compressing means, BW transform means and LZ compressing means. This test pattern compressing apparatus 21 comprises test pattern dividing means 402 for dividing an input data into blocks in accordance with a data structure or a statistical characteristic of the input data, run length compressing means 403 for applying a run length compression to a block having the number of data changes equal to or less than a threshold value, BW transform means 404 for applying BW transform to a block having the number of data changes greater than the threshold value and a data entropy equal to or less than a threshold value, another run length compressing means 405 for applying the run length compressing method to a data transformed by the BW transform means, and LZ compressing means 406 for applying an LZ compression to a block having the number of data changes greater than the threshold value and a data entropy greater than the threshold value.

[0143] Next, the operations for compressing a test pattern using this test pattern compressing apparatus 21 will be explained. Fig. 41 is a flow chart for explaining a processing procedure of the test pattern compressing apparatus 21 shown in Fig. 40 and shows a test pattern compressing method of the first embodiment of the present invention.

[0144] The test pattern dividing means 402 divides, in step 411, a test pattern data into a plurality of blocks in accordance with a data structure or a statistical characteristic of the test pattern data. The data of each block are transmitted via a switch to corresponding one of the run length compressing means 403, the BW transform means 404 and the LZ compressing means 406 each having an optimum compression algorithm for the corresponding block. Then those data are compressed, in steps 413, 414, 415 and 416 by the run length compressing means 403, the BW transform means 404, the run length compressing means 405 and the LZ compressing means 406, respectively. Finally, in step 417, the compressed data are outputted via a switch.

[0145] Fig. 42 is, in the first embodiment of the present invention, a block diagram showing an example of a construction of the test pattern expanding apparatus in the case where the test pattern expanding apparatus has run length expanding means, inverse BW transform means and LZ expanding means. This test pattern expanding apparatus 41 comprises compressed data dividing means 422 for dividing a compressed data to be expanded into data compressed by the run length compressing method, data compressed by the run length compressing method after application of BW transform, and data compressed by the LZ compressing method by, for example, a flag affixed to each compressed data; run length expanding means 423 and 424, and LZ expanding means 426 for expanding respective compressed data to the original test patterns without any information loss; and inverse BW transform means 425 for applying inverse BW transform to test pattern data to which BW transform has been applied.

[0146] Next, the operations for expanding a compressed data using this test pattern expanding apparatus 41 will be explained. Fig. 43 is a flow chart for explaining a processing procedure of the test pattern expanding apparatus shown in Fig. 42 and shows a test pattern expanding method of the first embodiment of the present invention.

[0147] First, in step 431, the compressed data dividing means 422 divides a compressed data to be expanded into data compressed by the run length compressing method, data compressed by the run length compressing method after application of BW transform, and data compressed by the LZ compressing method. The compressed data are transmitted, in step 432, to the run length expanding means 423 and 424, and the LZ expanding means 426 respectively via a switch. The transmitted data are expanded, in steps 433, 434 and 436, by the respective data expanding means to the original data without any information loss. Further, the data to which BW transform has been applied is inversely transformed, in step 435 to the original data by inverse BW transform. Finally, in step 437, the expanded test patterns are outputted via a switch.

[0148] Fig. 44 is a block diagram showing an example of a functional construction of the test pattern dividing means 422 used in the first embodiment of the present invention. This test pattern dividing means 422 comprises parameter input means 442 for inputting environmental parameters for a block division, parameter storage means 443 for storing inputted environmental parameters, threshold value calculating means of the number of data changes 444 for calculating a threshold value of the number of data changes for performing a block division using the environmental parameters stored in the parameter storage means 443, number of data changes counting means 445 for counting the number of data changes of an inputted data, number of data changes comparator means 446 for comparing the threshold value of the number of data changes with the number of data changes of an actual data, entropy threshold value calculating means 447 for calculating a threshold value of entropy for performing a block division using the environmental parameters stored in the parameter storage means 443, entropy measuring means 448 for measuring an entropy of an inputted data, entropy comparator means 449 for comparing the entropy threshold value with an entropy of an actual data, and block dividing means 440 for dividing a data into a plurality of blocks in accordance with the comparison results obtained by the number of data changes comparator means 446 and the entropy comparator means 449.

[0149] Next, the operations for dividing a test pattern into blocks using this test pattern dividing means 422 will be explained. Fig. 45 is a flow chart for explaining a processing procedure of the test pattern dividing means 422 shown in Fig. 44 and shows a test pattern dividing method of the first embodiment of the present invention.

[0150] First, in step 451, parameters for calculating a threshold value for a block division are inputted using the parameter input means 442. The inputted parameters are stored in the parameter storage means 443. Then in step 452, the threshold value calculating means of the number of data changes 444 calculates a threshold value of the number of data changes of a pattern for dividing an input test pattern into a plurality of blocks. Then in step 453, the number of data changes counting means 445 counts the number of data changes of a test pattern for each pin of the inputted test pattern.

[0151] The threshold value of the number of data changes calculated by the threshold value calculating means of the number of data changes 444 and the number of data changes obtained by the number of data changes counting means 445 are sent to the number of data changes comparator means 446 where the both data are compared with each other in step 454. At the same time, in step 455, the entropy threshold value calculating means 447 calculates an entropy threshold value for dividing the input test pattern into a plurality of blocks. Then in step 456, an appearing probability of each symbol in the input test pattern is measured. In addition, in step 457, an entropy of the input data is calculated from the probabilities obtained in the step 456. The above steps 456 and 457 are performed by the entropy measuring means 448. The threshold value calculated by the entropy threshold value calculating means 447 and an entropy obtained by the entropy measuring means 448 are sent to the entropy comparator means 449 where the both data are compared with each other in step 458. Here, the steps from the step 452 through the step 454 and the steps from the step 455 through the step 458 can be processed in parallel.

[0152] The comparison results obtained in the steps 454 and 458 are transmitted to the block dividing means 440. The block dividing means 440 first determines, in step 459, if the number of data change of a test pattern for each pin is equal to or less than the threshold value to classify, in step 4510, a test pattern having the number of data changes equal to or less than the threshold value to block 1. Further, regarding a test pattern having the number of data changes greater than the threshold value, a check is made, in step 4511 to see if its entropy is equal to or less than the threshold value. If the entropy of the test pattern is equal to or less than the threshold value, the test pattern is classified to block 2 in step 4512. A test pattern having an entropy greater than the threshold value is classified to block 3 in step 4513. Finally, the block dividing means 440 transmits the data of each divided block to corresponding one of the plurality of data compressing means.

[0153] Fig. 46 is a block diagram showing another example of a functional construction of a test pattern dividing means 422 used in the first embodiment of the present invention. This test pattern dividing means 422 comprises parameter input means 462 for inputting environmental parameters for a block division, parameter storage means 463 for storing inputted environmental parameters, number of data changes dividing means 464 for performing a block division in accordance with the number of data changes, and entropy dividing means 465 for performing a block division in accordance with an entropy.

[0154] Further, the number of data changes dividing means 464 comprises threshold value calculating means of the number of data changes 466, number of data changes counting means 467 for counting the number of data changes of an inputted data, number of data changes comparator means 468 for comparing the threshold value of the number of data changes with the number of data changes of an actual data, and block dividing means 469 for dividing an input data into a plurality of blocks in accordance with the comparison results obtained by the number of data changes comparator means 468. The entropy dividing means 465 comprises entropy threshold value calculating means 4610 for calculating an entropy threshold value for performing a block division using the environmental parameters stored in the parameter storage means 463, entropy measuring means 4611 for measuring an entropy of an inputted data, entropy comparator means 4612 for comparing the entropy threshold value with an entropy of an actual data, and block dividing means 4613 for dividing a data into a plurality of blocks in accordance with the comparison results obtained by the entropy comparator means 4612.

[0155] Next, the operations for dividing a test pattern into blocks using the test pattern dividing means 461 will be explained. Fig. 47 is a flow chart for explaining a processing procedure of the test pattern dividing means shown in Fig. 46 and shows a test pattern dividing method of the first embodiment of the present invention.

[0156] First, in step 471, parameters for calculating a threshold value for a block division are inputted using the parameter input means 462. The inputted parameters are stored in the parameter storage means 463. Then in step 472, the threshold value calculating means of the number of data changes 466 in the number of data changes dividing means 464 calculates, using the parameters stored in the parameter storage means 463, a threshold value of the number of data changes of a pattern for dividing an input test pattern into a plurality of blocks. Then in step 473, the number of data changes counting means 467 counts the number of data changes of a test pattern for each pin of the inputted test pattern.

[0157] The threshold value of the number of data changes calculated by the threshold value calculating means of the number of data changes 466 and the number of data changes obtained by the number of data changes counting means 467 are sent to the number of data changes comparator means 468, where the both data are compared with each other in step 474. The comparison result obtained in the step 474 is transmitted to the block dividing means 469. Then in step 475, the block dividing means 469 determines if the number of data changes of a test pattern for each pin is equal to or less than the threshold value to classify, in step 476, a test pattern having the number of data changes equal to or less than the threshold value to block A. The divided block A is outputted, in step 477, via a switch. A test pattern having the number of data changes greater than the threshold value is classified, in step 478, to block B and the divided block B is transmitted to the entropy dividing means 465 in step 479. Then in step 4710, the entropy threshold value calculating means 4610 calculates an entropy threshold value for dividing an input test pattern into a plurality of blocks. In step 4711, an appearing probability of each symbol in the inputted test pattern is measured. In addition, in step 4712, an entropy of the input data is calculated from the probabilities obtained in the step 4711. The above steps 4711 and 4712 are performed by the entropy measuring means 4611.

[0158] The threshold value calculated by the entropy threshold value calculating means 4610 and the entropy obtained by the entropy measuring means 4611 are sent to the entropy comparator means 4612, where the both data are compared with each other in step 4713. Here, each of the steps 472 and 4710 is independent and can be processed in parallel with each other.

[0159] The comparison result obtained in the step 4713 is transmitted to the block dividing means 4613. In the block dividing means 4613, a check is made in step 4714 to see if an entropy of a test pattern for each pin is equal to or less than the threshold value. If an entropy of a test pattern is equal to or less than the threshold value, the test pattern is classified to block B1 in step 4715. A test pattern determined to have an entropy greater than the threshold value in the step 4714 is classified, in step 4716, to block B2. Finally, the block dividing means 4613 outputs in step 4717 the data of each block divided by the entropy dividing means 465 via a switch.

[0160] Fig. 48 is a block diagram showing further another example of a functional construction of a test pattern compressing apparatus 21 used in the first embodiment of the present invention. This test pattern compressing apparatus 21 comprises compressing method determining means 124 for determining an optimum compressing method adapted to an inputted test pattern, run length compressing means 125 for performing a data compression by applying a run length compressing method to an input test pattern, repetitive BW transform means 126 for applying BW transform at least one time to an input test pattern, LZ compressing means 127 for performing a data compression by applying an LZ compressing method to an input test pattern, and three switches 128, 129 and 131 for selecting a rout of a test pattern in accordance with the compressing method obtained by the compressing method determining means 124.

[0161] Next, the operations for compressing a test pattern using this test pattern compressing apparatus will be explained. Fig. 49 is a flow chart for explaining a processing procedure of the test pattern compressing apparatus shown in Fig. 48 and shows a test pattern compressing method of the first embodiment of the present invention.

[0162] First, in step S1, the compressing method determining means 124 determines an optimum compressing method adapted to an inputted test pattern and sets a flag that can unitarily determine a compressing method in "kind". Here, the flag takes a value 0 for the run length compressing method, a value m (m=1, 2, • • • , N, where N is a fixed value indicating the maximum number of application times of BW transform, for example, N=5) which is equal to the number of application times of BW transform for the BWT run length compressing method, and a value N+1 for the LZ compressing method. Further, in order to determine an optimum compressing method to be applied to a test pattern, an experientially determined threshold value input is used.

[0163] Next, in steps S2a, S2b and S2c, when the determined compressing method is the run length compressing method (i.e., kind=0), switches 128, 129 and 131 are connected to their terminals 1, 1 and 1, respectively to transmit, in step S3, the test pattern to the run length compressing means 125. Then in step S4, the run length compressing means 125 compresses the test pattern using the run length compressing method.

[0164] In step S2b, when the determined compressing method is the BWT run length compressing method (i.e., 0<kind<N+1), the switches 128, 129 and 131 in the test pattern compressing apparatus are connected to their terminals 2, 2 and 1, respectively to transmit, in step S5, the test pattern and the number of application times of BW transform M (=kind) to the repetitive BW transform means 126. Then, in step S6, the repetitive BW transform means 126 applies BW transform to the test pattern M times. Further in step S7, the data transformed by BW transform is transmitted to the rum length compressing means 125, and then in step S8, the run length compressing means 125 compresses the data.

[0165] In step S2c, when the determined compressing method is the LZ compressing method (i.e., kind=N+1), the switches 128 and 131 in the test pattern compressing apparatus are connected to their terminals 3 and 2 respectively to transmit, in step S9, the test pattern to the LZ compressing means 127. Then in step S10, the LZ compressing means 127 compresses the test pattern using the LZ compressing method. Finally in step S11, the compressed data and the flag "kind" which unitarily indicates the method used for the compression are outputted and the process ends.

[0166] Fig. 50 is a block diagram showing an example of a functional construction of the repetitive BW transform means 126 in the test pattern compressing apparatus shown in Fig. 48. This repetitive BW transform means comprises BW transform means 133 for applying BW transform to an inputted test pattern, a counter 134 for counting the number of application times of BW transform, and two switches 135 and 136 for selecting the data path.

[0167] Next, the operations for performing BW transform of a data using this repetitive BW transform means 126 will be explained. Fig. 51 is a flow chart for explaining a processing procedure of the repetitive BW transform means 126 shown in Fig. 50.

[0168] First in step S1, an inputted test pattern is transmitted to the BW transform means 133 and the number of application times M of BW transform is transmitted to the counter 134. Then in step S2, the number of application times m of BW transform is set to m=M and a counter value i is reset to i=0. In step S3, BW transform is applied to the test pattern and the counter 134 adds 1 to i based on an end signal of BW transform. In step S4, a check is made to see if BW transform is performed m times, i.e., i=m. If the counter value of the number of applied times i of BW transform is not i=m, the switch 135 is opened and the switch 136 is connected to its terminal 1 to repeat BW transform in the step S3.

[0169] If the counter value of the number of applied times i of BW transform is i=m, the switch 135 is closed and the switch 136 is connected to its terminal 2 to output the data transformed by BW transform in next step S5, and the process ends.

[0170] Next, the operations for performing BW transform using the BW transform means 133 will be explained. Fig. 52 is a flow chart for explaining a processing procedure of the repetitive BW transform means shown in Fig. 50 and shows a BW transform method.

[0171] First, in step S1, a character string S having n characters is acquired. Here, an explanation will be made using, as an example, a character string S="abraca" having 6 characters (n=6). Then in step S2, n circulating character strings (cyclically shifted character strings) of the acquired character string S are generated. The circulating character strings are character strings wherein a first character of a character string is cyclically moved to the last position of the character string (shown in Fig. 53(a)). In the example S="abraca" used here, 6 circulating character strings are obtained as shown in Fig. 53(b).

[0172] Next, in step S3, the circulating character strings obtained in the step S2 are sorted in the order of registration into a dictionary to create a matrix M. In the example of S="abraca", the sorting is performed in this example, in alphabetical order like aa in first order, ab in second order, and ac in third order as shown in Fig. 54. In steps S4 and S5, a character string L obtained from the last column of the matrix M and a number I assigned to a row having the same character string as the input character string S are acquired. In the example of S="abraca", as shown in Fig. 55, L="caraab" and I=2 are obtained.

[0173] As explained above, the character string S="abraca" is transformed to a data containing a character string L="caraab" and a row number I=2. Finally, in step S6, a check is made to see if there are remaining input data. If there are remaining input data, the steps S1-S5 are repeated. If there is no remaining input data, the data transformed by BW transform are outputted and the process ends.

[0174] Fig. 56 is a block diagram showing an example of a processing procedure of a test pattern expanding apparatus used in the first embodiment of the present invention. This test pattern expanding apparatus comprises compressing method determining means 147 for determining whether a compressed data is a data compressed by a run length compressing method, or a data compressed by a BWT run length compressing method, or a data compressed by an LZ compressing method; run length expanding means 142 for expanding a compressed data using the run length expanding method; repetitive inverse BW transform means 43 for applying inverse BW transform at least one time to a data expanded by the run length expanding method; LZ expanding means 144 for expanding a compressed data using an LZ expanding method; and three switches 145, 146 and 147 for selecting a compressed data path in accordance with a compressing method obtained by the compressing method determining means 41.

[0175] Next, the operations for expanding a test pattern using this test pattern expanding apparatus will be explained. Fig. 57 is a flow chart for explaining a processing procedure of the test pattern expanding apparatus shown in Fig. 56 and shows a first embodiment of the test pattern expanding method of the present invention.

[0176] First, in step S1, the compressing method determining means 141 acquires a flag which is outputted from the test pattern compressing means together with a compressed data and unitarily indicates a compressing method, and sets this flag in "kind" as kind=flag. Here, the flag is determined to take a value 0 for the run length compressing method, a value m (m=1, 2, • • • , N, where N is a fixed value indicating the maximum number of application times of BW transform, for example, N=5) which is equal to the number of application times of BW transform for the BWT run length compressing method, and a value N+1 for the LZ compressing method.

[0177] Next, in steps S2a-S2c, the processing procedure is split into cases. When the determined compressing method is the run length compressing method (kind=0), the switches 145, 146 and 147 in the test pattern expanding apparatus are connected to their terminals 1, 1 and 1, respectively to transmit, in step 153, the compressed data to the run length expanding means 142, and then the compressed data is expanded, in step S4, by the run length expanding means 142 using the run length expanding method. In the step S2, when the determined compressing method is the BWT run length expanding method (0<kind<N+1), the switches 145, 146 and 147 in the test pattern expanding apparatus are connected to their terminals 1, 2 and 2, respectively, to transmit, in step S5, the compressed data to the run length expanding means 142, and then in step S6, the compressed data is expanded by the run length expanding means 142.

[0178] Next, in step S7, the expanded data and the number of application times M of inverse BW transform (=kind) are transmitted to the repetitive inverse BW transform means 143 and in step S8, the repetitive inverse BW transform means 143 applies inverse BW transform M times to the data. In the step S2, when the determined compressing method is the LZ compressing method (kind=N+1), the switches 145 and 147 in the test pattern expanding apparatus are connected to their terminal 2 and 3, respectively to transmit, in step S9, the compressed data to the LZ expanding means 144, and then in step S10, the compressed data is expanded by the LZ expanding means 144 using the LZ expanding method. Finally, in step S11, the expanded test pattern is outputted and the process ends.

[0179] Fig. 58 is a block diagram showing an example of a construction of the repetitive inverse BW transform means 143 in the test pattern expanding apparatus shown in Fig. 56. This repetitive inverse BW transform means 143 comprises inverse BW transform means 148 for applying inverse BW transform to an input data, a counter 149 for counting the number of application times of inverse BW transform, and two switches 151 and 152 for selecting a data path.

[0180] Next, the operations for performing inverse BW transform of a data using this repetitive inverse BW transform means 143 will be explained. Fig. 59 is a flow chart for explaining a processing procedure of the repetitive inverse BW transform means 143 shown in Fig. 58 and shows the operations for performing inverse BW transform of a data using the repetitive inverse BW transform means 143.

[0181] First in step S1, an inputted data is transmitted to the inverse BW transform means 148 and the number of application times M of inverse BW transform is transmitted to the counter 149. Then in step S2, the number of application times m of inverse BW transform is set to m=M and the counter 149 is reset to i=0. In step S3, inverse BW transform is applied to a test pattern and in response to an end signal of the inverse BW transform operation, the counter 149 adds one (1) to i. In step S4, a check is made to see if inverse BW transform is performed m times, i.e., i=m. If the number of application times i is not i=m, the switch 51 is opened and the switch 152 is connected to its terminal 1 to repeat inverse BW transform of the step S3. If the number of applied times i is i=m, the switch 151 is closed and the switch 152 is connected to its terminal 2. In next step S5, the transformed data by inverse BW transform is outputted and the process ends.

[0182] The other operations for performing inverse transform using the inverse BW transform means 143 will be further explained. Fig. 60 is a flow chart for explaining a processing procedure of the inverse BW transform means in the test pattern expanding apparatus shown in Fig. 56 and shows a method of inverse BW transform. Here, the explanation will be made using the case of L='caraab' and I=2 used in the explanation of BW transform.

[0183] First, in step S1, a transformed data is acquired. In step S2, a character string L, a number I and a length of the character string n obtained from the transformed data are set. In this example, those data are L='caraab', I=2 and n=6.

[0184] Next, in step S3, the characters in the acquired character string L are sorted in alphabetical order to generate a character string F. In the example of the character string L='caraab', F is F='aaabcr' as shown in Fig. 61. In step S4, a matrix T showing the correlations between a character L[i] of the character string L and a character F[i] of the character string F is created. Here, T is a matrix that satisfies $L[T[i]]=F[i]$. In addition, if the same characters ch are used in the character string L, those characters are correlated between the character string L and the character string F such that the characters ch in the character string L are in the same order as in the character string F. Therefore, in the example of L='caraab' and F='aaabcr', as shown in Fig. 61, T=[245613] is obtained.

[0185] Next, in step S5, the counter 149 is reset to i=0. In step S6, the counter is incremented by one (1). In step S7, an ith character S[i] in the original character string S is restored from the character string F, the number I and the matrix T. Here, S[i] is represented by $S[i]=F[T^{i-1}[I]]$, $T^{0}[I]=I$ and $T^{i+1}[I]=T^{i}[I]$. In step S8, a character S[i] is outputted. That is, when i is i=1 first, S[1] is $F[T^0[I]]$. As mentioned above, since $T^0[I]$ is $T^0[I]=I$ and I=2 in this example, S[1] is F(2). Since second character in F is a, S[1]=a is obtained. Then, when i is i=2, S[2] is $F[T^{1}[I]]=T[T^{0}[I]=T[2]$. Since second number in T is T=4, S[2] is F[4] Since fourth character in F is b, S[2]=b is obtained. When i is i=3, S[3] is $F[T^{2}[I]]=T[T^{1}[I]]=T[4]$. Since fourth number in T is T=6, S[3] is F[6]. Since sixth character in F is r, S[3]=r is obtained. In similar manner, inverse BW transform operations are performed hereinafter.

[0186] Next, in step S9, a check is made to see if there are characters not restored yet, i.e., i<n. If characters not restored are remaining (if i<n), the steps S6-S8 are repeated. If there is no character not restored (if i=n), the process proceeds to step S10.

[0187] In the example used here, as shown in Fig. 62, S[1]=a, S[2]=b, S[3]=r, S[4]=a, S[5]=c and S[6]=a are obtained. Therefore, the original character string S='abraca' is reconstructed. Finally, in step S10, a check is made to see if there are remaining data. If there are remaining data, the steps S1-S9 are repeated. If there is no remaining input data, the process ends.

[0188] Fig. 63 is a block diagram showing an example of a functional construction of the compressing method determining means 124 in the test pattern compressing apparatus shown in Fig. 48. This compressing method determining means comprises BW transform optimizing means 154 for calculating the optimum number of application times of BW transform for applying BW transform to a test pattern, threshold value storage means 155 for storing a threshold value of compression rate, compression rate comparator means 156 for comparing the maximum compression rate value obtained from the BW transform optimizing means 154 with the threshold value stored in the threshold value storage means 155, and compressing method selecting means 157 for selecting a compressing method out of the run length compressing method, the BWT run length compressing method and the LZ compressing method in accordance with the comparison result obtained from the compression rate comparator means 156 and the optimum number of application times of BW transform obtained from the BW transform optimizing means 154.

[0189] Next, the operations for determining the optimum compressing method to be applied to a test pattern using this compressing method determining means 124 will be explained. Fig. 64 is a flow chart for explaining a processing procedure of the compressing method determining means 124 shown in Fig. 63 and shows the operations of the compressing method determining method.

[0190] First, in step S1, the maximum application number n of an initialized BW transform is set to n=N (for example, N=5), a threshold value R_{th} of a compression rate is externally set to a certain value (for example $R_{th}=10$), and a counter i, the maximum value of the compression rate R_{max} and the number of application times n of BW transform are reset to all zeros to initialize environmental variables. In step S2, an inputted test pattern is transmitted to the BW transform optimizing means 154. Then in step S3, the test pattern is transformed once by BW transform and one (1) is added to the counter i. In step S4, the number of data changes ϕ_i of the data transformed by BW transform is measured.

[0191] Next, in step S5, a compression rate R_i is calculated from the obtained number of data changes ϕ_i and the number of applied times i of BW transform using the formula (1).

$$R_i=\log_2|A|/\lceil i\cdot\lceil\log_2 L\rceil+\phi_i\cdot\lceil\log_2\{(|A|-1)\cdot L\}+\log_2(L-1)+\log_2|A|\rceil\rceil \qquad (5)$$

Here, L:length of the test pattern, |A|:size of a set of symbols appearing in the test pattern, in this example, |A|=3 since the symbols are 0, 1 and X, $\lceil X\rceil$:the minimum integer equal to or greater than X.

[0192] In step S6, the maximum value of the compression rate R_{max} obtained by that time is compared with a newly calculated compression rate R_i. If the newly calculated compression rate R_i is greater than the obtained maximum value R_{max} ($R_i>R_{max}$), the maximum value of the compression rate R_{max} is updated to $R_{max}=R_i$ and the number of application times m of BW transform is updated to m=i at that time.

[0193] Next, in step S8, a check is made to see if the number of applied times of BW transform has become the number of times n specified in the step S1, i.e., i<n. If the number of applied times i is less than the specified number of times n, the steps S3-S7 are repeated. If the number of applied times i has become the specified number of times n, the process proceeds to step S9.

[0194] The steps S3-S8 are performed by the BW transform optimizing means 154. That is, the BW transform optimizing means 154 obtains, by repeating the steps S3-S8 by the specified number of times n, the maximum value of the compression rate R_{max} and the number of application times m of BW transform at that time. Then in step S9, the maximum value of the compression rate R_{max} obtained by the BW transform optimizing means 154 and the experimentally obtained threshold value R_{th} stored in the threshold value storage means 155 are transmitted to the compression rate comparator means 156 to compare them with each other. In step S10, if the maximum value of the compression rate R_{max} is equal to or less than the threshold value R_{th}, a compressing method to be applied is determined, in step 11, to be the LZ compressing method and a flag is set to
flag=N+1. If the threshold value R_{th} is less than the maximum value of the compression rate R_{max} in the step S10, a check is made in next step 12 to see if the number of application times m of BW transform is zero.

[0195] In the step 12, if the number of application times m of BW transform is zero, a compressing method to be applied is determined, in step 13, to be the run length compressing method and the flag is set to flag=0. If the number of application times m of BW transform is not zero, a compressing method to be applied to the test pattern is determined, in step 14, to be the BWT run length compressing method and the flag is set to flag=m. The steps S11-S14 are performed by the compressing method selecting means 157. Finally, in step S15, the compressing method selecting means 157 outputs a flag for unitarily determining the compressing method and the process ends.

[0196] In the above operations, the threshold value of the compression can be set either externally or to a predetermined fixed value. In the case of setting the threshold value externally, the threshold value storage means 155 operates as a RAM. On the other hand, in the case of setting the threshold value to the predetermined fixed value, the threshold value storage means 155 operates as a ROM and at this time, a threshold value input is not necessary.

[0197] The combination of the compressing methods which provides the most efficient results at present is the case in which three methods of the run length compressing method, the BW transform method plus the run length compressing method and the LZ compressing method are combined. A test pattern can be divided into three portions of a portion in which pattern changes are very few, a portion in which the pattern changes periodically, and a portion in which the pattern changes at random. An efficient compression can be obtained by applying the run length compressing method, the BW transform method plus the run length compressing method and the LZ compressing method to those three portions, respectively. Here, although the LZ compressing method is used for the portion in which the pattern changes at random, the Huffman compressing method or the like may also be used. Since compressing methods which can be used by anybody are mostly based on the LZ compressing method and the LZ compressing method provides a better compression rate, the LZ compressing method is used in the above example.

[0198] As mentioned above, the present invention is applied to a compression and to an expansion of a test pattern to be applied to an LSI for a test thereof. However, by dividing a document file (data) composed of, for example, text information, image information, layout information and the like into several data portions each having a different characteristic from each other, the compression rate can be improved. For example, a text compression technology such as the LZ compressing method is applied to a text data portion, an image compression technology such as JBIG or JPEG is applied to an image data portion, and the Huffman compressing method or the like is applied to a layout information portion.

[0199] Next a second embodiment of the present invention will be explained in detail.

[0200] Fig. 65 is a block diagram showing an example of a functional construction of a test pattern compressing apparatus used in the second embodiment of the present invention. This test pattern compressing apparatus is a test pattern compressing apparatus having one input and one output and comprises test pattern dividing means 11 for dividing an inputted test pattern data into test sequences each being for each pin, test sequence compressing means 12 for compressing a divided test sequence using an optimum compressing method, and a buffer 13 for temporarily storing a compressed data.

[0201] Next, the operations for performing a test pattern compression using this test pattern compressing apparatus will be explained. Fig. 66 shows a test pattern compressing method of the second embodiment of the present invention. The test pattern dividing means 11 divides, in step S1, an inputted test pattern into test sequences each being for each pin. Then in step S2, the number of divisions k of the test sequences is set to k=K and a counter i is reset to i=0. Here, K is, for example, the number of pins of an integrated circuit. In step S3, one test sequence is selected out of the divided test sequences and the counter i is incremented by one. Then in step S4, the selected test sequence is transmitted to test sequence compressing means 12. The test sequence compressing means 12 compresses, in step S5, the test sequence using an optimum method.

[0202] Here, in order to determine the optimum compressing method to be applied to each test sequence, a threshold value input is used. The compressed data are temporarily stored, in step S6, in a buffer 13 in order. Then in step S7, a check is made to see if there are remaining test sequences, i.e., if i is i<k. If there are remaining test sequences (if i is i<k) the steps S3-S6 are repeated. If there is no remaining test sequence (if i is i=k), the process moves to step S8. Finally, in step S8, the compressed data are collectively outputted and the process ends.

[0203] Fig. 67 is a block diagram showing a variation example of a construction of the test pattern compressing apparatus used in the second embodiment of the present invention. This test pattern compressing apparatus is a test pattern compressing apparatus having one input and one output, and comprises test pattern dividing means 11 for dividing an inputted test pattern data into test sequences each being for each pin and test sequence compressing means 12 for compressing a divided test sequence using an optimum compressing method.

[0204] Next, the operations for compressing a test pattern using this test pattern compressing apparatus will be explained. Fig. 68 shows a variation example of the test pattern compressing method of the second embodiment of the present invention. The test pattern dividing means 11 divides, in step S1, an inputted test pattern into test sequences each being for each pin. Then in step S2, the number of divisions k of the test sequences is set to k=K and a counter i is reset to i=0. Here, K is, for example, the number of pins of an integrated circuit. In step S3, a test sequence is selected out of the divided test sequences and the counter i is incremented by one. Then in step S4, the selected test sequence is transmitted to the test sequence compressing means 12. The test sequence compressing means 12 compresses, in step S5, the test sequence using an optimum method.

[0205] Here, in order to determine the optimum compressing method to be applied to each test sequence, a threshold value input is used. Then in step S7, the compressed data is outputted in the unit of test sequence. Finally, in step S7, a check is made to see if there are remaining test sequences, i.e., if i is i<k. If there are remaining test sequences (if i is i<k), the steps S3-S6 are repeated. If there is no remaining test sequence (if i is i=k), the process ends.

[0206] Fig. 69 is a block diagram showing an example of another functional construction of the test pattern compressing apparatus used in the second embodiment of the present invention. This test pattern compressing apparatus is a test pattern compressing apparatus having one input and multiple outputs, and comprises the test pattern dividing means 11 for dividing an inputted test pattern data into test sequences each being for each pin, a plurality of test pattern compressing means 12_1-12_n for compressing in parallel the divided test sequences using respective optimum compressing methods, and a switch 14 for selecting those test sequence compressing means 12_1-12_n.

[0207] Next, the operations for compressing a test pattern using this test pattern compressing apparatus will be explained. Fig. 70 shows another example of the test pattern compressing method of the second embodiment of the present invention. The test pattern dividing means 11 divides, in step S1, an inputted test pattern into test sequences each being for each pin. Then in step S2, the test pattern dividing means 11 transmits in order the divided test sequences to the plurality of test sequence compressing means 12_1-12_n respectively via the switch 14 and activates the test sequence compressing means 12_1-12_n.

[0208] Each of the test sequence compressing means 12_1-12_n compresses, in steps $S3_1$-$S3_n$, the corresponding test sequence using the optimum method. Here, in order to determine the optimum compressing method to be applied to each test sequence, a threshold value input is used. Finally, in steps $S4_1$-$S4_n$, the compressed data are outputted in parallel and the process ends. Since it takes a relatively long time to compress a test sequence, this parallel processing allows a high speed process.

[0209] Fig. 71 shows an example of a construction of a test pattern expanding apparatus used in the second embodiment of the present invention. This test pattern expanding apparatus is a test pattern expanding apparatus having one input and one output and comprises compressed data dividing means 16 for dividing a compressed data to be expanded into compressed data each being for each pin, test sequence expanding means 17 for expanding a divided compressed data to the original test sequence without any information loss, and a buffer 18 for temporarily storing an expanded test sequence.

[0210] Next, the operations for expanding a compressed data using this test pattern expanding apparatus will be explained. Fig. 72 is a flow chart for explaining a processing procedure of a test pattern expanding method of the second embodiment of the present invention.

[0211] First, in step S1, the compressed data dividing means 16 divides a compressed data to be expanded into compressed data each being for each pin. Then in step S2, the number of divisions k of the compressed data is set to k=K and a counter i is reset to i=0. Here, K is, for example, the number of pins of an integrated circuit. In step S3, a compressed data is selected out of the divided compressed data and the counter i is incremented by one. In step S4, the selected compressed data is transmitted to the test sequence expanding means 17. The test sequence expanding means 17 completely expands, in step S5, the compressed data to the original test sequence. The expanded data are stored, in step S6, in the buffer 18 in order. Then a check is made, in step S7, to see if there are remaining compressed data, i.e., if i is i<k. If there are remaining compressed data (if i is i<k), the steps S3-S6 are repeated. If there is no remaining compressed data (if i is i=k), the process moves to step S8. Finally, in step S8, the expanded test sequences are collectively outputted and the process ends.

[0212] Fig. 73 is a block diagram showing a variation example of a construction of the test pattern expanding apparatus used in the second embodiment of the present invention. This test pattern expanding apparatus is a test pattern expanding apparatus having one input and one output and comprises compressed data dividing means 16 for dividing a compressed data to be expanded into compressed data each being for each pin and test sequence expanding means 17 for expanding a compressed data to the original test sequence without any information loss.

[0213] Next, the operations for expanding a compressed data using this test pattern expanding apparatus will be explained. Fig. 74 shows a variation example of the test pattern expanding method of the second embodiment of the present invention.

[0214] First, in step S1, the compressed data dividing means 16 divides a compressed data to be expanded into compressed data each being for each pin. Then in step S2, the number of divisions k of the compressed data is set to k=K and a counter i is reset to i=0. Here, K is, for example, the number of pins of an integrated circuit. In step S3, one compressed data is selected out of the divided compressed data and the counter i is incremented by one. Then in step S4, the selected compressed data is transmitted to the test sequence expanding means 17. The test sequence expanding means 17 completely expands, in step S5, the compressed data to the original test sequence. Then in step S6, the expanded test sequence is outputted in each pin basis. Finally, in step S7, a check is made to see if there are remaining compressed data, i.e., if i is i<k. If there are remaining compressed data (if i is i<k), the steps S3-S6 are repeated. If there is no remaining compressed data (if i is i=k), the process ends.

[0215] Fig. 75 is a block diagram showing another functional construction example of the test pattern expanding apparatus used in the second embodiment of the present invention. This test pattern expanding apparatus is a test pattern expanding apparatus having one input and multiple outputs and comprises compressed data dividing means 16 for dividing a compressed data to be expanded into compressed data each being for each pin, a plurality of test sequence expanding means 17_1-17_n for expanding the respective divided compressed data in parallel to the original test sequences without any information loss, and a switch 19 for selecting the test sequence expanding means.

[0216] Next, the operations for expanding a compressed data using this test pattern expanding apparatus will be explained. Fig. 76 shows a test pattern expanding method of the second embodiment of the present invention.

[0217] First, in step S1, the compressed data dividing means 16 divides a compressed data to be expanded into compressed data each being for each pin. Then in step S2, the divided compressed data are transmitted to the plurality of test sequence expanding means 17_1-17_n respectively and activates those test sequence expanding means 17_1-17_n. Each of the test sequence expanding means 17_1-17_n completely expands the corresponding one of the compressed data to the original test sequence in steps $S3_1$-$S3_n$. Finally, in steps $S4_1$-$S4_n$, the expanded test sequences are outputted in parallel and the process ends. A data expansion can be performed in a short time period compared with a data compression but requires a high speed processing. By processing in parallel in pin basis in this manner, the expanding means each having low processing speed can be used.

[0218] Fig. 77 is a block diagram showing an example of a construction of a test pattern expanding apparatus used in the second embodiment of the present invention. This test pattern expanding apparatus is a test pattern expanding apparatus having multiple inputs and one output and comprises a plurality of test sequence expanding means 17_1-17_n connected in parallel for expanding compressed data each being for each pin inputted in parallel to the original test sequences respectively without any information loss, test sequence buffers 520_1-520_n for temporarily storing therein the expanded test sequences respectively, a switch 521 for selecting one of those test sequence buffers 520_1-520_n to output corresponding one of the expanded test sequences, and a buffer 522 for temporarily storing therein all the expanded test sequences.

[0219] Next, the operations for expanding a compressed data using this test pattern expanding apparatus will be explained. Fig. 78 shows a test pattern expanding method of the second embodiment of the present invention.

[0220] Compressed data each being for each pin inputted in parallel are transmitted to the plurality of test sequence expanding apparatus, respectively. First, in step S1, each of the test sequence expanding means 17_1-17_n is activated. Then in step S2, each of the test sequence expanding means 17_1-17_n completely expands corresponding one of the compressed data to the original test sequence and temporarily stores it in corresponding one of the test sequence buffers 520_1-520_n. The expanded test sequences are stored, in step S3, in the buffer 522. Here, those test sequences can be stored, in response to expansion end signals from the test sequence expanding means 17_1-17_n, in the buffer 522 in fixed order after all the expansions of the test sequences are completed, or can be stored in the buffer 522 in the sequence of the expansion completion. Finally, in step S4, a test pattern is composed of the expanded test sequences from the buffer 522 and is outputted. Then the process ends.

[0221] Fig. 79 shows a variation example of a construction of the test pattern expanding apparatus used in the second embodiment of the present invention. This test pattern expanding apparatus is a test pattern expanding apparatus having multiple inputs and one output and comprises a plurality of test sequence expanding means 17_1-17_n connected in parallel for expanding compressed data each being for each pin inputted in parallel to the original test sequences respectively without any information loss, test sequence buffers 520_1-520_n for temporarily storing therein the expanded test sequences respectively, and a switch 521 for selecting one of those test sequence buffers 520_1-520_n to output corresponding one of the expanded test sequences.

[0222] Next, the operation for expanding a compressed data using this test pattern expanding apparatus will be explained. Fig. 80 shows a variation example of the test pattern expanding method of the second embodiment of the present invention.

[0223] Compressed data each being for each pin inputted in parallel are transmitted to the plurality of test sequence expanding apparatus, respectively. First, in step S1, each of the test sequence expanding means 17_1-17_n is activated. Then in steps $S2_1$-$S2_n$, each of the test sequence expanding means 17_1-17_n completely expands corresponding one of the compressed data to the original test sequence and stores it in corresponding one of the test sequence buffers 520_1-520_n. Finally, in steps $S3_1$-$S3_n$, the expanded test sequences are outputted in respective pin basis and the process ends.

[0224] Here, those test sequences can be outputted, in response to expansion end signals from the test sequence expanding means 17_1-17_n, in fixed order after all the expansions of the test sequences are completed, or can be outputted in the sequence of the expansion completion. In either of the constructions shown in Figs. 77 and 79, by making the outputted expanded test sequences same as the test pattern before compression, those test sequences can be supplied to a pattern generator conventionally used in an IC tester.

[0225] Fig. 81 shows an example of a functional construction of a test sequence compressing apparatus used in the second embodiment of the present invention. This test sequence compressing apparatus comprises compressing method determining means 524 for determining an optimum compressing method adaptive to an inputted test sequence, run length compressing means 525 for compressing a test sequence using the run length compressing method, repetitive BW transform means 526 for applying BW transform one or more times to a test sequence, LZ compressing means 527 for compressing a test sequence using a LZ compressing method, and three switches 528, 529 and 531 for selecting a test sequence path in accordance with the compressing method obtained by the compressing method determining means 524.

[0226] Next, the operations for compressing a test sequence using this test sequence compressing apparatus will be explained. Fig. 82 shows a test sequence compressing method of the second embodiment of the present invention.

[0227] First, in step S1, the compressing method determining means 524 determines an optimum compressing method adaptive to an inputted test sequence and sets a flag for unitarily determining the compressing method in "kind". Here, the flag takes a value of 0 for the run length compressing method, a value of m (m=1, 2, • • •, N where N is a fixed value indicating the maximum number of application times of BW transform: for example, N=5) for the BWT run length compressing method, and a value of N+1 for the LZ compressing method. In addition, in order to determine an optimum compressing method to be applied to a test sequence, an input of a threshold value which is experimentally determined is used.

[0228] Next, in steps S2a, S2b and S2c, if the determined compressing method is the run length compressing method (i.e., kind=0), the switches 528, 529 and 531 of the test sequence compressing apparatus are connected to terminals 1, 1, 1, respectively, and in step S3, the test sequence is transmitted to the run length compressing method 525. Then in step S4, the run length compressing means 525 compresses the test sequence using the run length compressing method. If, in the step S2b, the determined compressing method is the BWT run length compressing method (i.e., 0<kind<N+1), the switches 528, 529 and 531 are connected to terminals 2, 2 and 1, respectively, and in step S5, the test sequence and the number of application times M (=kind) of BW transform are transmitted to the repetitive BW transform means 526. Then in step S6, the repetitive BW transform means 526 applies BW transform M times to the test sequence. Further, in step S7, the data transformed by BW transform is transmitted to the run length compressing means 525 where, in step S8, the data is compressed. If, in the step S2c, the determined compressing method is the LZ compressing method (i.e., kind=N+1), the switches 528 and 531 of the test sequence compressing apparatus are connected to terminals 3 and 2, respectively and the test sequence is transmitted, in step S9, to the LZ compressing means 527. Then in step S10, the LZ compressing means 527 compresses the test sequence using the LZ compressing method. Finally, in step S11, the compressed data and the flag "kind" unitarily indicating the method used in the compression are outputted and the process ends.

[0229] Fig. 83 shows an example of a functional construction of the repetitive BW transform means 526 in the test sequence compressing means shows in Fig. 81. This repetitive BW transform means comprises BW transform means 533 for applying BW transform to an inputted test sequence, a counter 534 for counting the number of application times of BW transform, and two switches 535 and 536 for selecting a data path.

[0230] Next, the operations for performing BW transform of a data using this repetitive BW transform means 526 will be explained. Fig. 84 is a flow chart showing the operations of BW transform of a data performed by the repetitive BW transform means.

[0231] First, in step S1, an inputted test sequence is transmitted to the BW transform means 533 and the number of application times M of BW transform is transmitted to the counter 534. Then in step S2, the number of application times m of BW transform is set to m=M and the counter i is reset to i=0. In step S3, the test sequence is transformed by BW transform and the counter 534 adds one to the counter value i in response to an end signal of BW transform. In step S4, a check is made to see if BW transform is performed m times, i.e., if i is i=m. If the number of applied times i of BW transform is not i=m, the switch 535 is opened and the switch 536 is connected to terminal 1 to repeat BW transform of the step S3. If the number of applied times i is i=m, the switch 535 is closed and the switch 536 is connected to terminal 2. In next step S5, the data transformed by BW transform is outputted and the process ends.

[0232] Here, the run length compression, the BW transform and the LZ compression will be explained.

[0233] First, the operations for compressing a data using the run length compressing means 525 will be explained. Fig. 85 is a flow chart showing the run length compressing method. First, in step S1, one character data is acquired from an input character string and is set in ch. Then in step S2, the acquired character ch is set to an initial value of symbol "initial" for generating run length codes and a counter i is initialized to one (1). Then in step S3, one character is acquired from the input data and is set in ch and then in step S4, one (1) is added to the counter i to count the number of characters. In step S5, the acquired character ch is compared with the initial value "initial". If the character ch is equal to the initial value "initial" (ch=initial), the process proceeds to step S6. If the character ch is different from the initial value "initial" (ch≠initial), the process proceeds to step S7.

[0234] In step S6, a check is made to see if there are remaining input data. If there are remaining input data, the steps S3, S4 and S5 are repeated. If there is no remaining input data, the process proceeds to step S7. In step S7, an encoding is performed using the initial value "initial" and the count value i. Then in step S8, a check is made to see if there are remaining input data. If there are remaining input data, the steps S2-S7 are repeated. If there is no remaining input data, the compressed data are outputted in step S9, and the process ends.

[0235] Next, the operations for transforming a data using the BW transform means 533 will be explained. Fig. 86 is a flow chart showing the BW transform method. First, in step S1, a character string S having n characters is acquired. In this case, an explanation is made using, as an example, a character string S= 'abraca' having n=6 characters. Then in step S2, n cyclic character strings (cyclically shifted character strings) of the acquired character string S are generated. The cyclic character strings are character strings in which the first character of a character string is cyclically moved to the last character position of the character string (shown in Fig. 87(a)). In the example of S='abraca' used in this case, six cyclic character strings are obtained as shown in Fig. 87(b). Then in step S3, the cyclic character strings obtained in the step S2 are sorted in the order of registration into a dictionary to create a matrix M. In the example of S='abraca', the character strings are sorted in the alphabetic order as shown in Fig. 88 in such a manner as aa first, ab next and ac next.

[0236] In steps S4 and S5, a character string L obtained from the last column of the matrix M and a row number I of a row having the same character string as the input character string S are acquired from the created matrix M. In the example of S="abraca", as shown in Fig. 89, L='caraab' and I=2 are obtained. As mentioned above, the character string S='abraca' is transformed to a data containing a character string L='caraab' and a row number I=2. Finally, in step S6, a check is made to see if there are remaining input data. If there are remaining input data, the steps S1-S5 are repeated. If there is no remaining input data, the data transformed by BW transform are outputted, in step S7, and the process ends.

[0237] Finally, the operations for compressing a data using the LZ compressing means 527 will be explained. Fig. 90 is a flow chart showing the LZ compressing method. First, in step S1, an initialization of a dictionary dic to be used in a data compression is performed. Then in step S2, an input data "in" is acquired and in step S3, a matching between the acquired character string "in" and the dictionary dic is performed. In step S4, a check is made to see if there is a character string that matches better with the dictionary dic. If there is a possibility that there may be a character string matching better with the dictionary dic, the matching operation in the step S3 is repeated. If there is no character string that matches better with the dictionary dic in step S4, the character string "match" matched with the dictionary dic and the first character ch which did not match with the dictionary dic are coded in step S5. In step S6, this new character string match + ch are registered in the dictionary. Finally, in step S7, a check is made to see if there are remaining input data. If there are remaining input data, the steps S2-S6 are repeated. If there is no remaining input data, the compressed data are outputted in step S8 and the process ends.

[0238] Fig. 91 is a block diagram showing an example of a functional construction of the test sequence expanding means used in the second embodiment of the present invention. This test sequence expanding means comprises compressing method determining means 541 for determining whether a compressed data has been compressed by the run length compressing method, by the BWT run length compressing method, or by the LZ compressing method; run length expanding means 542 for expanding a compressed data using the run length expanding method; repetitive inverse BW transform means 543 for applying inverse BW transform to a data expanded by the run length expanding means one or more times; LZ expanding means 544 for expanding a compressed data using the LZ expanding method; and three switches 545, 546 and 547 for selecting a compressed data path in accordance with the compressing method obtained by the compressing method determining means 541.

[0239] Next, the operations for expanding a test sequence using this test sequence expanding apparatus will be explained. Fig. 92 is a flow chart showing the test sequence expanding method of the second embodiment of the present invention. First, in step S1, the compressing method determining means 541 acquires a flag which is outputted together with a compressed data from the test sequence compressing means and unitarily indicates the compressing method to set this flag in "kind" as kind=flag.

[0240] Here, the flag is determined to take a value of 0 for the run length compressing method, a value of the number of application times m (M=1, 2, • • • , N where N is a fixed value indicating the maximum number of application times of BW transform, for example, N=5) of BW transform for the BWT run length compressing method, or a value of N+1 for the LZ compressing method. Then in steps S2a-S2c, different operations are performed in accordance with the compressing method "kind". If the determined compressing method is the run length compressing method (kind=0), the switches 545, 546 and 547 of the test sequence expanding means are connected to a terminal 1, a terminal 1 and a terminal 1, respectively. In step S3, a compressed data is transmitted to the run length expanding means 542 and then in step S4, the run length expanding means 542 expands the compressed data using the run length expanding method.

[0241] If, in step S2, the determined compressing method is the run length compressing method (0<kind<N+1), the switches 545, 546 and 547 of the test sequence expanding apparatus are connected to their terminals 1, 2 and 2, respectively. In step S5, the compressed data is transmitted to the run length expanding means 542 and the run length expanding means 542 expands, in step S6, the compressed data. Then in step S7, the expanded data and the number of application times M (=kind) of inverse BW transform are transmitted to the repetitive inverse BW transform means 543 and the repetitive inverse BW transform means 543 inversely transforms the data M times, in step S8, the by inverse BW transform.

[0242] If, in step S2, the determined compressing method is the LZ compressing method (kind=N+1), the switches 545 and 547 of the test sequence expanding means are connected to the terminals 2 and 3, respectively. In step S9, the compressed data is transmitted to the LZ expanding means 544 and the LZ expanding means 544 expands, in step S10, the compressed data using the LZ expanding method. Finally, in step S11, the expanded test sequence is outputted and the process ends.

[0243] Fig. 93 is a block diagram showing an example of a construction of the repetitive inverse BW transform means 543 in the test sequence expanding means shown in Fig. 91. This repetitive inverse BW transform means 543 comprises inverse BW transform means 548 for applying inverse BW transform to an inputted data, a counter 549 for counting the number of application times of inverse BW transform, and two switches 551 and 552 for selecting a data path.

[0244] Next, the operations for performing inverse BW transform of a data using this repetitive inverse BW transform means 543 will be explained. Fig. 94 is a flow chart showing the operations of inverse BW transform of a data by the repetitive inverse BW transform means 543. First, in step S1, an input data is transmitted to the inverse BW transform means 548 and the number of application times M of inverse transform is transmitted to the counter 549. Then in step S2, the number of application times m of inverse BW transform is set to m=M and the counter 549 is reset to i=0. In step S3, a test sequence is transformed by inverse BW transform and the counter 549 adds one (1) to i in response to an end signal of an inverse BW transform operation. In step S4, a check is made to see if inverse BW transform is performed m times, i.e., if i is i=m. If the number of applied times i is not i=m, the switch 551 is opened and the switch 552 is connected to terminal 1, and then the inverse BW transform operation of the step 3 is repeated. If the number of applied times i is i=m, the switch 551 is closed and the switch 552 is connected to a terminal 2. Then in next step S5, the data transformed by inverse BW transform is outputted and the process ends.

[0245] Here, the run length expansion, the inverse BW transform and the LZ expansion will be explained.

[0246] First, the operations for expanding a data using the run length expanding means 542 will be explained. Fig. 95 is a flow chart showing the run length expanding method. First, in step S1, a compressed data is acquired. In step S2, an initial value "initial" and the number of character continuation times n are acquired from the compressed data. Then in step S3, the initial value "initial" is copied by n times (the number of character continuations). In step S4, a check is made to see if there are remaining compressed data. If there are remaining compressed data, the steps S1-S3 are repeated. If there is no remaining data, the expanded data are outputted in step S5 and the process ends.

[0247] Next, the operations for performing inverse transform using the inverse BW transform means 543 will be explained.

[0248] Fig. 96 is a flow chart showing the inverse BW transform method. Here, an explanation will be given using the example of L='caraab' and I=2 used in the explanation of BW transform. First, in step S1, a transformed data is acquired and in step S2, the character string L and the number I obtained from the transformed data and the length of the character string n are set. In this example, L='caraab', 1=2 and n=6 are set. Then in step S3, the characters in the acquired character string L are sorted in alphabetical order to generate a character string F. In the example of L='caraab', F is, as shown in Fig. 97, F='aaabcr'. In step S4, a matrix T indicating the correlations between each character L [i] in the character string L and each character F [i] in the character string F is created. Here, T is a matrix which satisfies $L[T[i]]=F[i]$. In addition, if the same characters ch are used in the character string L, those characters ch in the character string L are sorted in the character string F such that the sequence of those characters ch are unchanged. Therefore, in the example of L='caraab' and F= 'aaabcr', T is obtained as T=[245613] as shown in Fig. 97.

[0249] Next, in step S5, the counter 549 is reset to i=0. In step S6, the counter 549 is incremented by one (1) and in step S7, ith character S[i] of the original character string S is restored from the character string F, the number I and the matrix T. Here, S[i] is represented by $S[i]=F[T^{i-1}[I]]$, $T^{0}[I]=I$, $T^{i+1}[I]=T[T^{i}[I]]$. In step S8, the character S[i] is outputted.

That is, when i is $i=1$ first, $S[1]=F[T^0[l]]$. Since, as mentioned above, $T^0[l]=l$ and $l=2$ in this example, $S[1]=F[2]$ is obtained. Therefore, the second character a of F is $S[1]=a$. Next, when i is $i=2$, $S[2]=F[T^1[l]]$ is obtained. Since $T^1[l]=T[T^0[l]]=T[2]$ and the second number of T is 4, $S[2]=F[T[2]]=F[4]$ is obtained. Since the fourth character of F is b, $S[2]=b$ is obtained. When i is $i=3$, $S[3]$ is $F[T^2[l]]$. Since $T^2[l]$ is $T^2[l]=T[T^1[l]]=T[4]$, the fourth number of T is 6. Therefore, $S[3]$ is $F(6)$. Since the sixth character of F is r, $S[3]$ is $S[3]=r$. Hereinafter, inverse BW transform is similarly performed.

[0250] Then in step S9, a check is made to see if there are remaining characters not restored yet in the character string S, i.e., if i is $i<n$. If there are remaining characters not restored yet (if i is $i<n$), the steps 6-8 are repeated. If the characters have been restored up to the last character (i is $i=n$), the process proceeds to step S10. In the example used here, as shown in Fig. 98, $S[i]$ is obtained as; $S[1]=a$, $S[2]=b$, $S[3]=r$, $S[4]=a$, $S[5]=c$ and $[6]=a$. Thus, the original character string S='abraca' has been reconstructed. Finally, in step S10, a check is made to see if there are remaining data. If there are remaining data, the steps S1-S9 are repeated. If there is no remaining data, the process ends.

[0251] Finally, the operations for expanding a data using the LZ expanding means 544 will be explained. Fig. 99 is a flow chart showing the LZ expanding method. First, in step S1, a dictionary dic to be used in the expansion process is initialized. Then in step S2, one code is acquired from an input compressed data. In step S3, an information retrieval is performed in the dictionary using the acquired data. In step S4, the code is converted to a character string "string". Then in step S5, the newly generated character string "string" is registered in the dictionary dic. Finally, a check is made, in step S6, to see if there are remaining compressed data. If there are remaining compressed data, the steps S2-S5 are repeated. If there is no remaining compressed data, the expanded data are outputted in step S7 and the process ends.

[0252] Fig. 100 is a block diagram showing an example of a functional construction of the compressing method determining means 524 in the test sequence compressing means shown in Fig. 81. This compressing method determining means 524 comprises BW transform optimizing means 554 for calculating the optimum number of application times for applying BW transform to a test sequence, threshold value storage means 555 for storing a threshold value of compression rate, compression rate comparator means 556 for comparing the maximum value of compression rate obtained from the BW transform optimizing means 544 with the threshold value stored in the threshold value storage means 555, and compressing method selecting means 557 for selecting whether a test sequence is compressed by the run length compressing method, by the BWT run length compressing method, or by the LZ compressing method in accordance with the comparison result obtained from the compression rate comparator means 556 and the optimum number of application times of BW transform obtained from the BW transform optimizing means 554.

[0253] Next, the operations for determining the optimum compressing method to be applied to a test sequence using this compressing method determining means 524 will be explained.

[0254] Fig. 101 is a flow chart showing this compressing method determining method. First, in step S1, the initialized maximum application times n of BW transform is set to $n=N$ (for example, $N=5$), a threshold value of compression rate R_{th} is externally set to a certain value (for example, $R_{th}=10$), and each of a counter i, the maximum value of compression rate R_{max} and the number of application times n of BW transform is reset to zero to initialize environmental variables. In step S2, an inputted test sequence is transmitted to the BW transform optimizing means 554. Then in step S3, BW transform is applied to the test sequence one time and the counter i is incremented by one. In step S4, the number of data changes ϕ_i of the data transformed by BW transform is measured.

[0255] Next, in step S5, a compression rate R_i is calculated using the formula (4) from the obtained number of data changes ϕ_i and the number of applied times i of BW transform. In step S6, the maximum value R_{max} of the compression rates obtained before now is compared with the newly calculated compression rate R_i. If the newly calculated compression rate R_i is greater than the maximum value R_{max} (if R_i is $R_i > R_{max}$), in step S7, the maximum value of compression rate R_{max} is updated to $R_{max}=R_i$ and the number of application times m of BW transform is updated to $m=i$ at that time.

[0256] Next, in step S8, a check is made to see if the number of applied times of BW transform has become the number n specified in the step 1, i.e., if i is $i<n$. If the number of applied times i is less than the specified number n, the steps S3-S7 are repeated. If the number of applied times i has become the specified number n, the process proceeds to step S9. The steps S3-S8 are performed by the BW transform optimizing means 554. That is, the BW transform optimizing means 554 acquires, by repeating the steps S3-S8 by the specified number of times n, the maximum value R_{max} and the number of application times m of BW transform at that time. Then in step S9, the maximum value of compression rate R_{max} obtained by the BW transform optimizing means 554 and the threshold value of compression rate R_{th} stored in the threshold value storage means 555 are transmitted to the compression rate comparator means 556 to be compared with each other.

[0257] In step S10, if the maximum value R_{max} of compression rate is equal to or less than the threshold value R_{th}, the compressing method to be applied is determined, in step S11, to be the LZ compressing method and a flag is set to flag=N+1. If, in the step S10, the threshold value R_{th} is less than the maximum value R_{max}, the number of application times m of BW transform is confirmed in step S12.

[0258] If, in step S12, the number of application times m of BW transform is zero, the compressing method to be applied to the test sequence is determined, in step S13, to be the run length compressing method and the flag is set to flag=0. If the number of application times m of BW transform is not zero, the compressing method to be applied to the test sequence is determined, in step S14, to be the BWT run length compressing method and the flag is set to flag=m. The steps S11-S14 are performed by the compressing method selecting means 557. Finally, the compressing method selecting means 557 outputs, in step S15, the flag which unitarily determines a compressing method and then the process ends.

[0259] In the above operations, the threshold value R_{th} of compression rate can either externally be set or be set in advance to a fixed value. When the threshold value R_{th} is externally set, the threshold value storage means 555 operates as a RAM. On the other hand, when the threshold value R_{th} is set in advance to a fixed value, the threshold value storage means 555 operates as a ROM and in this case, the threshold value input is not necessary.

[0260] Fig. 102 is a block diagram showing an example of a construction of the BWT run length compressing means. This BWT run length compressing means comprises repetitive BW transform means 561 for applying BW transform one or more times to a test sequence and run length compressing means 562 for compressing a data transformed by BW transform using the run length compressing method.

[0261] Next, the operations for compressing a test sequence without any information loss using this BWT run length compressing means will be explained. Fig. 103 is a flow chart showing the BWT run length compressing method. First, in step S1, an inputted test sequence and the number of application times M (=kind) of BW transform are transmitted to the repetitive BW transform means 561. Then in step S2, the number of application times m of BW transform is set to m=M and a counter i is reset to i=0. In step S3, the test sequence is transformed by BW transform and one (1) is added to the counter i. In step S4, a check is made to see if BW transform is performed m times, i.e., if i is i=m. The step 3 is repeated until the number of applied times i becomes m. Then in step S5, the data transformed m times by BW transform is transmitted to the run length compressing means 562 and the data is compressed, in step S6, using the run length compressing method. Finally, in step S7, the compressed data is outputted and the process ends.

[0262] Fig. 104 is a block diagram showing an example of a construction of the BWT run length expanding means. This BWT run length expanding means comprises run length expanding means 563 for expanding an inputted compressed data using the run length expanding method and repetitive inverse BW transform means 564 for applying inverse BW transform one or more times to the data expanded by the run length expanding means.

[0263] Next, the operations for expanding a compressed data to the original test sequence without any information loss using this BWT run length expanding means will he explained. Fig. 105 is a flow chart showing the BWT run length expanding method. First, in step S1, an inputted compressed data is transmitted to the run length expanding means 563. Then in step S2, the run length expanding means 563 expands the compressed data. In step S3, the expanded data and the number of application times M (=kind) of inverse BW transform are transmitted to the repetitive inverse BW transform means 564. Then in step S4, the number of application times m of inverse BW transform is set to m=M and a counter i is reset to i=0. The repetitive inverse BW transform means 564 inversely transforms, in step S5, the data by inverse BW transform and the counter i is incremented by one (1). Then in step S6, a check is made to see if inverse BW transform is performed m times, i.e., if i is i<m. The step S5 is repeated until the number of applied times of inverse BW transform becomes m. Finally, in step S7, the reconstructed test sequence is outputted and the process ends.

[0264] Heretofore, the test pattern compressing method, the test pattern expanding method, the test pattern compressing apparatus and the test pattern expanding apparatus of the second embodiment of the present invention have been explained. However, the present invention further includes an automatic IC test system which can transfer a test pattern at high speed using the test pattern compressing apparatus or the test pattern compressing method and the test pattern expanding apparatus or the test pattern expanding method according to the present invention. The automatic IC test system (IC tester) having the test pattern expanding apparatus according to the present invention can cope with various test environments such as an environment in which a pattern generator in the main body of the tester is connected to a disk drive storing test patterns via a tester processor, or an environment in which a pattern generator in the main body of the tester is directly connected to a disk array apparatus or the like. There will be explained below with reference to Figs. 106-113 the embodiments and the variation examples of the automatic IC test system having a test pattern expanding apparatus and the embodiments and the variation examples of the automatic IC test system having a test pattern compressing apparatus.

[0265] Fig. 106 is a block diagram showing a construction of a first embodiment of an automatic IC test system having a test pattern expanding apparatus and a test pattern compressing apparatus according to the present invention. This test system comprises a tester processor 570 for controlling a main body of the tester 579, a circuit for storing test data and for generating, in synchronism with a basic clock, logical data to be applied to a semiconductor integrated circuit under test (CUT) 571, namely a pattern generator 572 for reading out and outputting in order the stored logical data, a buffer memory 573 which is a circuit for making up for a memory provided in the pattern generator 572 and can perform a high speed data transfer with the pattern generator 572, a test pattern expanding apparatus 574 having one input and one output for expanding a compressed data at high speed and for storing the expanded data in the pattern generator 572, format controllers 575 each being provided for each pin for generating, from logical data generated by the pattern generator 572, pulses having different widths or pulses having phase differences from each other, pin electronics 576 each being provided for each pin and having a driver for applying an electric signal to the CUT 571 and a comparator to which a response from the CUT 571 is supplied, an interface 577 for interfacing with the CUT 571, and a workstation 578 for controlling the main body of the tester.

[0266] The controlling workstation 578 is used for operating the main body of the tester 579 to control the automatic test of the IC 571. For example, a SPARC computer of Sun Microsystems Inc. can be used as the workstation 578. Test patterns have been compressed by a test pattern compressing apparatus 581 and have already been stored in a disk drive 582 of the workstation 578.

[0267] In a test of the CUT 571, when a test pattern is down-loaded into the pattern generator 572 of the main body of the tester, a compressed test pattern data is transferred to the test pattern expanding apparatus 574 via the workstation 578 and the tester processor 570, and then the compressed test pattern data is expanded at high speed and the expanded test pattern data is stored in the pattern generator 572. If the entire test pattern data is not accommodated in the pattern generator 572, the remaining portion of the test pattern data is stored, in compressed state, in the buffer memory 573 and is expanded by the test pattern expanding apparatus 574 when necessary. In this case, if the test pattern is arranged in the same arrangement as in that of a conventional test pattern, a conventional pattern generator can be used as the pattern generator 572. By using the above construction, the data quantity or amount passing through a data transferring path can be reduced and the test pattern down-loading time can be reduced.

[0268] Fig. 107 is a block diagram showing a construction of a second embodiment of the automatic IC test system having the test pattern expanding apparatus according to the present invention. This test system comprises a tester processor 570 for controlling a main body of the tester 579, a pattern generator 572 which is a circuit for storing test data and for generating, in synchronism with basic clock, logical data to be applied to a semiconductor integrated circuit under test (CUT) 571, a test pattern expanding apparatus 574 having one input and one output for expanding a compressed data at high speed and for storing the expanded data in the pattern generator 572, a disk array apparatus 585 which stores therein test pattern data and is connected to a network 584, and can transfer a test pattern data at high speed, format controllers 575 each being provided for each pin for generating, from logical data generated by the pattern generator 572, pulses having different widths or pulses having phase differences from each other, pin electronics 576 each being provided for each pin and having a driver for applying an electric signal to the CUT 571 and a comparator to which a response from the CUT 571 is supplied, an interface 577 for interfacing with the CUT 571, and a workstation 578 for controlling the main body of the tester.

[0269] The controlling workstation 578 is used for operating the main body of the tester 579 to control the automatic test of the IC 571. For example, a SPARC Computer of Sun Microsystems Inc. can he used as the workstation 578. Test patterns are compressed by a test pattern compressing apparatus (not shown) and are stored in the disk array apparatus 585 via the network 584. In a test of the CUT 571, when a test pattern stored in the disk array apparatus 585 is down-loaded into the pattern generator 572 of the main body of the tester 579, a compressed test pattern data is transferred to the test pattern expanding apparatus 574 from the disk array apparatus 585, and then the compressed test pattern data is expanded at high speed and the expanded test pattern data is stored in the pattern generator 572.

[0270] By using the above construction, the data quantity passing through a data transferring path can be reduced. As a result, the test pattern down-loading time can be reduced.

[0271] Next, a measuring system of the main body of the tester will be explained in detail. Fig. 108 shows an example of a detail construction of the measuring system of the main body of the tester 579 used in the embodiment shown in Fig. 106. The measuring system used in the above embodiment comprises a timing generator 572a for generating a timing pulse for defining a time period such as a delay, a width or the like, a pattern generator 572b for generating an expanded logical data in synchronism with a basic clock, format controllers 575 each being provided for each pin for generating, from the logical data generated by this pattern generator 572b and the clock signal from the timing generator 572a, pulses having different widths or pulses having phase differences from each other, pin electronics 576 each being provided for each pin and having a driver for applying an electric signal to the CUT 571 and a comparator to which a response from the CUT 571 is supplied, and an interface 577 for interfacing with the CUT 571.

[0272] In the test of the CUT 571, the pattern generator 572b generates a logical data to be applied to the CUT 571 in accordance with the clock generated by the timing generator 572a. Each of the format controllers 575 forms a pulse based on the logical data. The pulse generated by each of the format controllers 575 is converted to a signal having a level set by the driver of the corresponding pin electronics 576, and then the signal is applied to the CUT 571 via the interface 577. A response from the CUT 571 is acquired by the comparator of the corresponding pin electronics 576, where the response is compared with an expected value data outputted from the pattern generator 572b.

[0273] Fig. 109 is a block diagram showing a construction of a third embodiment of the automatic IC test system having the test pattern expanding apparatus and the test pattern compressing apparatus according to the present invention. This test system comprises a tester processor 570 for controlling a main body of the tester 579, a circuit for storing test data and for generating, in synchronism with a basic clock, logical data to be applied to a semiconductor integrated circuit under test (CUT) 571, namely a pattern generator 572 for reading out and outputting the stored data, a buffer memory 573 which is a circuit for making up for a memory provided in the pattern generator 572 and can perform a high speed data transfer with the pattern generator 572, test sequence expanding apparatus 574 each being provided for each pin for expanding compressed data outputted from the pattern generator 572 in parallel for respective pins at high speed, format controllers 575 each being provided for each pin for generating, from logical data generated by the pattern generator 572, pulses having different widths or pulses having phase differences from each other, pin electronics 576 each being provided for each pin and having a driver for applying an electric signal to the CUT 571 and a comparator to which a response from the CUT 571 is supplied, an interface 577 for interfacing with the CUT 571, and a workstation 578 for controlling the main body of the tester.

[0274] The controlling workstation 578 is used for operating the main body of the tester 579 to control the automatic test of the IC 571. For example, a SPARC computer of Sun Microsystems Inc. can be used as the workstation 578. Test patterns have been compressed by a test pattern compressing apparatus 581 and have already been stored in a disk drive 582 of the workstation 578.

[0275] In a test of the CUT 571, when a test pattern is down-loaded into the pattern generator 572 of the main body of the tester 579, a compressed test pattern data is stored in the pattern generator 572 via the workstation 578 and the tester processor 570. If the entire test pattern data is not accommodated in the pattern generator 572, the remaining portion of the test pattern data is stored, in compressed state, in the buffer memory 573 and is transferred to the pulse generator 572 when necessary. In addition, the compressed data outputted from the pattern generator 572 are transferred to the respective test sequence expanding apparatus 574 in the respective pin basis and are expanded at high speed, and then are sent to the respective format controllers 575.

[0276] By using the above construction, the data quantity passing through a data transferring path can be reduced. Hence, the test pattern down-loading time can be reduced.

[0277] Fig. 110 is a block diagram showing a construction of a fourth embodiment of the automatic IC test system having the test pattern expanding apparatus according to the present invention. This test system comprises a tester processor 570 for controlling a main body of the tester 579, a pattern generator 572 which is a circuit for storing test data and for generating, in synchronism with a basic clock, logical data to be applied to a semiconductor integrated circuit under test (CUT) 571, test pattern expanding apparatus 574 each being provided for each pin for expanding compressed data outputted from the pattern generator 572 in parallel for respective pins at high speed, a disk array apparatus 585 which stores therein test pattern data and is connected to a network 584, and can transfer a test pattern data at high speed, format controllers 575 each being provided for each pin for generating, from logical data generated by the pattern generator 572, pulses having different widths or pulses having phase differences from each other, pin electronics 576 each being provided for each pin and having a driver for applying an electric signal to the CUT 571 and a comparator to which a response from the CUT 571 is supplied, an interface 577 for interfacing with the CUT 571, and a workstation 578 for controlling the tester.

[0278] The controlling workstation 578 is used for operating the main body of the tester 579 to control the automatic test of the IC 571. For example, a SPARC computer of Sun Microsystems Inc. can be used as the workstation 578. Test patterns have been compressed by a test pattern compressing apparatus (not shown) and have already been stored in a disk array apparatus 585 via the network 584.

[0279] In a test of the CUT 571, when a test pattern stored in the disk array apparatus 585 is down-loaded into the pattern generator 572 of the main body of the tester 579, a compressed test pattern data is stored in the pattern generator 572 from the disk array apparatus 585. The compressed data outputted from the pattern generator 572 are transferred to the test sequence expanding apparatus 574 in the respective pin basis, where those compressed data are expanded at high speed and sent to the respective format controllers 575.

[0280] By using the above construction, the data quantity passing through a data transferring path can he reduced. Hence, the test pattern down-loading time can be reduced.

[0281] Next, a measuring system of the main body of the tester will be explained in detail. Fig. 111 shows an example of a detail construction of the measuring system of the main body of the tester 579 used in the embodiment shown in Fig. 109 or Fig. 110. The measuring system used in the above embodiments comprises a timing generator 572a for generating timing pulses for defining a time period such as a delay, a width or the like, a pattern generator 572b for generating an expanded logical data in synchronism with a basic clock, test sequence expanding apparatus each being provided for each pin for expanding in parallel the compressed data in the respective pin basis, format controllers 575 each being provided for each pin for generating, from the logical data generated by this pattern generator 572b and the clock signal from the timing generator 572a, pulses having different widths or pulses having phase differences from each other, pin electronics 576 each being provided for each pin and having a driver for applying an electric signal to the CUT 571 and a comparator to which a response from the CUT 571 is supplied, and an interface 577 for interfacing with the CUT 571.

[0282] In the test of the CUT 571, a test sequence expanding apparatus 574 generates, in accordance with a clock generated by the timing generator 572a, a logical data to be applied to the CUT 571 based on a compressed data outputted from the pattern generator 572b. Based on the logical data, each of the format controllers 575 forms a pulse. The pulse generated by each of the format controllers 575 is converted to an electric signal by the driver of the pin electronics 576, and then the signal is applied to the CUT 571 via the interface 577. A response from the CUT 571 is acquired by the comparator of the corresponding pin electronics 576, where the response is compared with an expected value data also outputted from the test sequence expanding apparatus 574.

[0283] By using the above construction, a compressed test pattern can be expanded in real time and the CUT 571 can be tested in real time using the expanded logical data. Further, by storing the compressed data in the pattern generator 572b, an amount of memory can be reduced.

[0284] Fig. 112 is a block diagram showing a construction of a fifth embodiment of the automatic IC test system having the test pattern compressing apparatus according to the present invention. This system includes a test pattern compressing apparatus 581 connected to a CPU memory bus 587 of a computer 601. A test pattern transferred from another computer via a network 584 is sent to the test pattern compressing apparatus 581 via an I/O controller 588, an I/O bus 589, a bus adapter 591 and the CPU memory bus 587. The test pattern is compressed by the test pattern compressing apparatus 581 without any information loss and the compressed data is stored, via the CPU memory bus 587, the bus adapter 591, the I/O bus 589 and the I/O controller 592, in a disk drive 582 or in a disk drive of another computer connected to the network 584. In addition, if the test pattern resides in a main memory 593, the test pattern is sent to the test pattern compressing apparatus 581 via the CPU memory bus 587. The test pattern is compressed by the test pattern compressing apparatus 581 without any information loss and is stored, via the CPU memory bus 587, the bus adapter 591, the I/O bus 589 and the I/O controller 592, in the disk drive 582 or in a disk drive of another computer connected to the network 584.

[0285] The compressed data stored in the disk drive 582 is transferred to the main body of the tester 579 via the I/O controllers 592 and 593 and the I/O bus 589. The test pattern to be transferred is already compressed and requires an only short transfer time. Therefore, if the compressed data can be expanded at high speed in the main body of the tester 579, an availability of the test system can be improved.

[0286] Fig. 113 is a block diagram showing a construction of a sixth embodiment of the automatic IC test system having the test pattern compressing apparatus according to the present invention. This system includes a test pattern compressing apparatus 581 disposed immediately before a disk drive 582. A test pattern transferred from another computer via a network 584 is sent to the test pattern compressing apparatus 581 via I/O controllers 588 and 592 and an I/O bus 589. Then the test pattern is compressed by the test pattern compressing apparatus 581 without any information loss and is stored in the disk drive 582. In addition, if the test pattern resides in a main memory 593, this test pattern is sent to the test pattern compressing apparatus 581 via the CPU memory bus 587, the bus adapter 591, the I/O bus 589 and the I/O controller 592. The test pattern is compressed by the test pattern compressing apparatus 581 without any information loss and the compressed data is stored in the disk drive 582. The compressed data stored in the disk drive 582 is transferred to the main body of the tester 579 via the I/O controller 592 and the I/O bus 589. The test pattern to be transferred is already compressed and requires an only short transfer time. Therefore, if the compressed data can be expanded at high speed in the test system, an availability of the test system can be improved.

[0287] In the case described above, the data compression is performed using the LZ compressing method when the compression rate is greater than the threshold value. However, there is no doubt that the other compressing methods such as the Huffman compressing method, the arithmetic coding compressing method or the like may be used.

[0288] According to the present invention, since each of received input data patterns is distributed to one of a plurality of blocks in accordance with its data structure or its statistical characteristic and a proper compressing method is applied to each block, a data comprised of data patterns having different data structures can efficiently be compressed.

[0289] In addition, according to the present invention, since an inputted test pattern data is divided into test sequences each being for each pin and a proper compressing method is applied to each test sequence, an efficient data compression of a test pattern data is made possible.

[0290] Fig. 114 shows results of the compression rate measurements when test patterns being used in an actual device test are compressed using the test pattern compressing method and the test pattern compressing apparatus according to the present invention. As samples of the test patterns, five test patterns (test sequence length is 15000) are used. In the illustration, K1 and K2 are test patterns for a CISC microcomputer (100 pins), A1 and A2 are test patterns for a disk controller (144 pins) and S1 is a test pattern for a RISC microcomputer (144 pins). Fig. 114 shows the comparisons with LZW (a compressing method used in UNIX compress etc.) which is a most generally used compressing method. It is shown that the test pattern compressing method and the test pattern compressing apparatus of the present invention provide better compression rates except for the test pattern of K1. Particularly for the test pattern of S1, a compression rate of more than 1500:1 is obtained by the present invention.

[0291] In addition, by using the test pattern expanding apparatus of the present invention, a high speed data expansion is possible in the automatic test system for automatically testing a semiconductor integrated circuit. As a result, a time period required for downloading a test pattern can be reduced.

[0292] Fig. 115 shows measurement results of expansion speeds of compressed data in the test pattern expanding apparatus of the present invention. There are shown in Fig. 115 comparisons between the expansion speeds in the test pattern expanding apparatus of the present invention and those similarly measured by the LZW compressing method.

From this illustration, it can be seen that the test pattern expanding apparatus of the present invention can provide a data expansion speed approximately two times higher than the conventional LZW compressing method.

[0293] The present invention can efficiently compress, using the BWT run length compressing apparatus, an M sequence (Maximum length sequence) which is one of Pseudorandom Noise sequences (PN sequences). Fig. 116 shows compression rate results when an M sequence is compressed using the BWT run length compressing apparatus. In the illustration, the results when BW transform is applied one time are indicated by a symbol x and the results when BW transform is applied two times are indicated by a symbol o. Further, a dashed line indicates the upper limit value of compression rate that can be obtained by the BWT run length compressing apparatus.

[0294] As shown in Fig. 116, by applying BW transform two times to an M sequence using the BWT run length compressing apparatus according to the present invention, a compression rate close to the upper limit can be obtained. Further, the compression rate is improved as the number of stages of a Linear Feedback Shift Register (LFSR) that generates an M sequence is increased and a compression rate of approximately 700:1 is obtained at the maximum.

[0295] In such a way, a high compression rate can be obtained by applying, after BW transform is applied a plurality of times, the run length compressing method to a data.

Claims

1. A data pattern compressing and expanding method including the steps of:

   dividing an inputted data pattern in accordance with its data structure to distribute each of the divided portions into one of a plurality of compressing blocks;
   compressing each of data divided into said plurality of compressing blocks using a compressing method suitable for each block;
   dividing the compressed data compressed in said compressing step into a plurality of expanding blocks in accordance with the compressing method of the data; and
   expanding the compressed data of each expanding block by a expanding method corresponding to the compressing method of the compressed data.

2. A data pattern compressing and expanding method including the steps of:

   dividing an inputted data pattern in accordance with its statistical characteristic to distribute each of the divided portions into one of a plurality of compressing blocks;
   compressing each of data divided into said plurality of compressing blocks using a compressing method suitable for each block;
   dividing the compressed data compressed in said compressing step into a plurality of expanding blocks in accordance with the compressing method of the data; and
   expanding the compressed data of each expanding block by a expanding method corresponding to the compressing method of the compressed data.

3. A data pattern compressing and expanding method including the steps of:

   counting the number of data changes of an inputted data pattern;
   measuring an appearing probability of each symbol in the inputted data pattern and calculating an entropy of the data based on the appearing probabilities of the symbols;
   comparing the number of data changes of the inputted data pattern with a threshold value of the number of data changes and comparing the entropy of the data with the threshold value of entropy;
   dividing the inputted data pattern in accordance with the comparison results to distribute each of the divided portions into one of a plurality of compressing blocks;
   compressing each of data divided into said plurality of compressing blocks using a compressing method suitable for each block;
   dividing the compressed data compressed in said compressing step into a plurality of expanding blocks in accordance with the compressing method of the data; and
   expanding the compressed data of each expanding block by an expanding method corresponding to the compressing method of the compressed data.

4. A data pattern compressing method for compressing an inputted data pattern including the steps of:

   dividing the inputted data pattern in accordance with its data structure to distribute each of the divided portions

into one of a plurality of blocks;
compressing each of data divided into said plurality of blocks using a compressing method suitable for each block.

5. The data pattern compressing method according to claim 4, wherein said dividing step includes the steps of:

counting the number of data changes of said input pattern data; and
comparing said counted number of data changes with a threshold value; and
wherein said dividing step further includes the step of distributing the data pattern to one of a plurality of blocks in accordance with the comparison result.

6. The data pattern compressing method according to claim 5, wherein said compressing step is the step of applying a run length compressing method to a block in which the number of data changes is less than the threshold value and applying another compressing method to a block in which the number of data changes is greater than the threshold value.

7. The data pattern compressing method according to claim 4, further including the steps of applying Burrows Wheeler transform (hereinafter referred to as BW transform) to a block having a large periodicity of data in the divided blocks and the run length compressing method is applied in said compressing step to the data transformed by BW transform.

8. A data pattern compressing method for compressing an inputted data pattern including the steps of:

dividing the inputted data pattern in accordance with a statistical characteristic of the data to distribute each of the divided portions into one of a plurality of blocks;
compressing each of data divided into said plurality of blocks using a compressing method suitable for each block.

9. The data pattern compressing method according to claim 8, wherein said dividing step includes the steps of:

measuring an appearing probability of each symbol in the inputted data pattern and calculating an entropy of the data from the appearing probabilities of the symbols;
comparing the entropy of the data with a threshold value; and
distributing the data in the inputted data pattern to one of a plurality of compressing blocks in accordance with the comparison results.

10. The data pattern compressing method according to claim 9, wherein a Huffman compressing method is applied in said compressing step to a block having an entropy less than the threshold value among the divided blocks.

11. The data pattern compressing method according to claim 9, wherein an LZ compressing method is applied in said compressing step to a block having an entropy less than the threshold value among the divided blocks.

12. The data pattern compressing method according to claim 9, wherein an arithmetic coding compressing method is applied in said compressing step to a block having an entropy less than the threshold value among the divided blocks.

13. A data pattern compressing method for compressing an inputted data pattern including the steps of:

counting the number of data changes of an inputted data pattern;
measuring an appearing probability of each symbol in the inputted data pattern and calculating an entropy of the data based on the appearing probabilities of the symbols;
comparing the number of data changes of the inputted data pattern with a threshold value of the number of data changes and comparing the entropy of the data with the threshold value of entropy;
dividing the inputted data pattern in accordance with the comparison results to distribute each of the divided portions into one of a plurality of compressing blocks; and
compressing each of data divided into said plurality of compressing blocks using a compressing method suitable for each block.

14. The data pattern compressing method according to claim 13, wherein in said compressing step, a data compression is performed by applying the run length compressing method to a block having the number of data changes equal to or less than the threshold value, and by applying the run length compressing method after performing BW transform at least one time to a block having the number of data changes greater than the threshold value of the number of data changes and an entropy equal to or less than the threshold value of entropy.

15. The data pattern compressing method according to any one of claims 5, 9 and 13, further including the steps of:

inputting environmental parameters; and
calculating the threshold values using the inputted environmental parameters.

16. The data pattern compressing method according to any one of claims 5, 9 and 13, further including the steps of:

calculating a threshold value for each of data patterns which are possible to occur; and
estimating a compression rate of the data pattern using the calculated threshold value to optimize the threshold value so as to maximize the compression rate; and
wherein the optimized threshold value is used in said comparing step.

17. The data pattern compressing method according to claim 4, wherein said dividing step comprises the steps of:

applying BW transform to an input data pattern;
counting the number of data changes of the data transformed by BW transform;
calculating a compression rate from the number of data changes;
obtaining the maximum value of compression rate and the number of application times of BW transform when the maximum value of compression rate is obtained;
comparing the maximum value of compression rate with a threshold value of compression rate; and
dividing the input data pattern to distribute each of the divided portions into one of a plurality of blocks.

18. The data pattern compressing method according to claim 17, wherein said compressing step comprises the steps of:

applying a compressing method such as an LZ compressing method, a Huffman compressing method, or an arithmetic coding compressing method to a block having compression rate equal to or greater than the threshold value; and
applying a run length compressing method to a block having compression rate equal to or less than the threshold value after performing BW transform the number of times by which the maximum compression rate is obtained.

19. The data pattern compressing method according to any one of claims 4 to 18, wherein the input data pattern is a pattern corresponding to each pin in a test pattern of a semiconductor integrated circuit.

20. A data pattern expanding method for expanding an inputted compressed data including the steps of:

dividing the compressed data into a plurality of blocks in accordance with a compressing method of the inputted compressed data; and
expanding each of the data divided into a plurality of blocks by a expanding method corresponding to a compressing method of each block.

21. The data pattern expanding method according to claim 20, wherein said dividing step is a step for dividing the inputted compressed data into data compressed by a run length compressing method and data compressed by the other compressing methods, and wherein the data compressed by the run length compressing method are expanded, in said expanding step, by a run length expanding method.

22. The data pattern expanding method according to claim 20, wherein in said dividing step, the inputted compressed data is divided into data directly compressed by a run length compressing method and data compressed by the run length compressing method after application of BW transform, and wherein in said expanding step, the data directly compressed by the run length compressing method are expanded by the run length expanding method to be outputted, and the data compressed by the run length compressing method after application of BW transform are

expanded by the run length expanding method and thereafter the expanded data are transformed by inverse BW transform to be outputted.

23. The data pattern expanding method according to claim 20, wherein in said dividing step, if there are data compressed by an LZ compressing method in the inputted compressed data, those data are separated from the other compressed data, and wherein in said expanding step, the data compressed by the LZ compressing method are expanded by an LZ expanding method.

24. The data pattern expanding method according to claim 20, wherein in said dividing step, the inputted compressed data is divided into data compressed by a Huffman compressing method and data compressed by the other compressing methods, and wherein in said expanding step, a Huffman expanding method is applied to the data compressed by the Huffman compressing method.

25. The data pattern expanding method according to claim 20, wherein in said dividing step, the inputted compressed data is divided into data compressed by an LZ compressing method and data compressed by the other compressing methods, and wherein in said expanding step, an LZ expanding method is applied to the data compressed by the LZ compressing method.

26. The data pattern expanding method according to claim 20, wherein in said dividing step, the inputted compressed data is divided into data compressed by an arithmetic coding compressing method and data compressed by the other compressing methods, and wherein in said expanding step, an arithmetic coding expanding method is applied to the data compressed by the arithmetic coding compressing method.

27. The data pattern expanding method according to claim 20, wherein in said dividing step, the inputted compressed data is divided into data directly compressed by a run length compressing method, data compressed by the run length compressing method after application of BW transform, and data compressed by the other compressing methods, and wherein in said expanding step, a run length expanding method is applied to the data directly compressed by the run length compressing method to output the expanded data, the run length expanding method is applied to the data compressed by the run length compressing method after application of BW transform and thereafter inverse BW transform is applied to the expanded data the number of times that BW transform was applied thereto at the compression time to output the expanded data, and an expanding method corresponding to one of the other compressing methods is applied to each of the data compressed by the other compressing methods to output the expanded data.

28. The data pattern compressing method according to any one of claims 20 to 27, wherein the inputted compressed data is a compressed data of a test pattern of a semiconductor integrated circuit and the expanded data outputted in said expanding step are patterns of the test pattern each corresponding to each pin.

29. A test pattern compressing apparatus comprising:

a plurality of compressing means each having a different compressing method from one another;
dividing means for dividing an inputted test pattern into test sequences each being for each of pins of an integrated circuit under test; and
compressing method determining means for selecting suitable compressing means out of a plurality of compressing means in accordance with a data structure of each test sequence to compress the corresponding test sequence in the basis of each pin.

30. The test pattern compressing apparatus according to claim 29, wherein said compressing method determining means and said plurality of compressing means are provided for each pin of said integrated circuit under test, and said dividing means supplies each test sequence to compressing method determining means of a corresponding pin.

31. The test pattern compressing apparatus according to claim 29 or 30, wherein said plurality of compressing means comprise run length compressing means for compressing a test sequence by a run length compressing method, BW transform means for applying BW transform at least one time to the test sequence, and BWT run length compressing means for supplying the transformed output of BW transform to said run length compressing means.

32. The test pattern compressing apparatus according to claim 29 or 30, wherein said plurality of compressing means

comprise compressing means other than said run length compressing means and said BWT run length compressing means.

33. The test pattern compressing apparatus according to claim 31 or 32, wherein said compressing method determining means comprises:

BW transform optimizing means for calculating the appropriate number of application times for applying BW transform to the inputted test sequence;
threshold value storage means for storing a threshold value of compression rate;
compression rate comparator means for comparing maximum value of compression rate obtained from said BW transform optimizing means with the threshold value stored in said threshold value storage means; and
compressing method selecting means for selecting a compressing method to compress a test sequence in accordance with the comparison result obtained from said compression rate comparator means and the appropriate number of application times BW transform obtained from said BW transform optimizing means.

34. A test pattern compressing method including the steps of:

dividing an inputted test pattern into test sequences each being for each of pins of an integrated circuit under test; and
selecting an appropriate compressing method in accordance with a data structure of each divided test sequence to compress the test sequence in the basis of each pin.

35. A test pattern compressing method including the steps of:

dividing an inputted test pattern into test sequences each being for each of pins of an integrated circuit under test; and
selecting appropriate compressing methods in parallel in accordance with respective data structures of the divided test sequences to compress the respective test sequences in the basis of respective pins.

36. The test pattern compressing method according to claim 34 or 35, further including the steps of:

determining an appropriate compressing method adaptive for an inputted test sequence; and
applying to the test sequence a run length compressing method or the run length compressing method after applying BW transform to the test sequence one or more times (hereinafter referred to as BWT run length compressing method), or one of the other compressing methods such as an LZ compressing method, a Huffman compressing method, an arithmetic coding compressing method or the like to compress the test sequence.

37. The test pattern compressing method according to claim 36, further including the steps of:

applying BW transform to the test sequence;
measuring the number of data changes of the data transformed by BW transform;
calculating a compression rate from the number of data changes;
obtaining the maximum value of compression rate and the number of application times of BW transform for maximizing the compression rate;
comparing the maximum value of compression rate with a threshold value of compression rate; and
selecting in accordance with the comparison result whether the test sequence is to be compressed by the run length compressing method, the BWT run length compressing method, or one of the other compressing methods.

38. A test pattern expanding apparatus comprising:

compressed data dividing means for dividing an inputted data into compressed data each being for each pin of an integrated circuit under test; and
a plurality of test sequence expanding means each for selecting an expanding method in accordance with a flag indicating a compressing method to apply the expanding method to each compressed data.

39. The test pattern expanding apparatus according to claim 38, wherein

said plurality of test sequence expanding means are provided for each pin of the integrated circuit under test;
said compressed data dividing means supplies each of the divided compressed data to corresponding one of said test sequence expanding means; and
said plurality of test sequence expanding means operate in parallel.

40. The test pattern expanding apparatus according to claim 38 or 39, further including:

compressing method determining means for determining, by the flag of the compressed data, whether the compressing method is a run length compressing method, a BWT run length compressing method, or one of the other compressing methods;
run length expanding means for expanding the compressed data using the run length expanding method;
repetitive inverse BW transform means for applying inverse BW transform one or more times to a data expanded by said run length expanding apparatus, and
the other expanding means for expanding the compressed data using the other expanding methods.

41. A test pattern expanding method including the steps of:

dividing an inputted compressed data into compressed data each being for each pin of an integrated circuit under test; and
expanding each divided compressed data by an expanding method corresponding to the compressing method specified by a flag of the compressed data.

42. A test pattern expanding method according to claim 41, wherein said expanding step is performed such that the compressed data are expanded in parallel in the basis of respective pins.

43. The test pattern expanding apparatus according to claim 41 or 42, wherein the expanding method used in said expanding step is a run length expanding method or an expanding method in which inverse BW transform is applied to the data expanded by the run length expanding method.

44. The test pattern expanding apparatus according to claim 43, wherein the other expanding methods are further included in the expanding methods used in said expanding step.

45. A test pattern compressing and expanding apparatus comprising:

a plurality of compressing means each having a different compressing method from one another;
dividing means for dividing an inputted test pattern into test sequences each being for each of pins of an integrated circuit under test;
compressing method determining means for selecting suitable compressing means out of a plurality of compressing means in accordance with a data structure of each test sequence to compress the corresponding test sequence in the basis of a pin;
compressed data dividing means for dividing an inputted data into compressed data each being for each pin of an integrated circuit under test; and
a plurality of test sequence expanding means each for selecting an expanding method in accordance with a flag indicating a compressing method to apply the expanding method to the corresponding compressed data.

46. A test pattern compressing and expanding method including the steps of:

dividing an inputted test pattern into test sequences each being for each of pins of an integrated circuit under test;
selecting an appropriate compressing method in accordance with a data structure of each divided test sequence data to compress the test sequence in the basis of a pin.
dividing an inputted compressed data into compressed data each being for each pin of an integrated circuit under test; and
expanding each divided compressed data by an expanding method corresponding to the compressing method specified by a flag of the compressed data.

47. A test pattern compressing and expanding method including the steps of:

dividing an inputted test pattern into test sequences each being for each of pins of an integrated circuit under test;
selecting appropriate compressing methods in parallel in accordance with respective data structures of the divided test sequences to compress the respective test sequences in the basis of respective pins;
dividing an inputted compressed data into compressed data each being for each pin of an integrated circuit under test; and
expanding each divided compressed data by an expanding method corresponding to the compressing method specified by a flag of the compressed data.

48. An automatic semiconductor integrated circuit test system for automatically testing a semiconductor integrated circuit comprising:

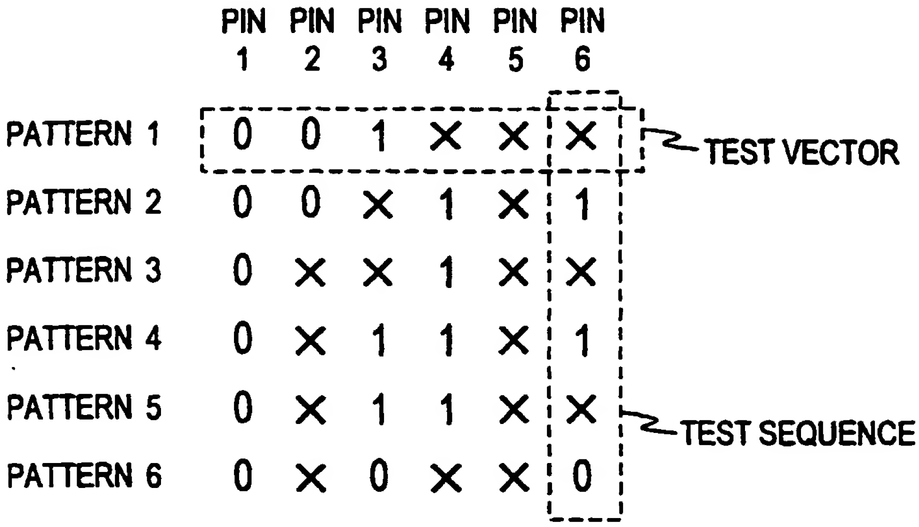
a test pattern compressing apparatus recited in claim 29;
a large capacity storage apparatus for storing compressed test pattern data;
a pattern generator for storing the test pattern data and for generating a test pattern; and
test pattern expanding means for restoring a compressed test pattern data to an original test pattern without any information loss.

49. The automatic semiconductor integrated circuit test system according to claim 48, wherein said test pattern expanding means is comprised of a plurality of test sequence expanding means for restoring test sequence data each being compressed for each pin in parallel in the basis of respective pins.

50. An automatic semiconductor integrated circuit test system for automatically testing a semiconductor integrated circuit comprising:

means for compressing a test pattern;
a large capacity storage apparatus for storing compressed test pattern data;
a pattern generator for storing the test pattern data and for generating a test pattern; and
a test pattern compressing and expanding apparatus recited in claim 45.

FIG. 1



	PIN 1	PIN 2	PIN 3	PIN 4	PIN 5	PIN 6
PATTERN 1	0	0	1	×	×	×
PATTERN 2	0	0	×	1	×	1
PATTERN 3	0	×	×	1	×	×
PATTERN 4	0	×	1	1	×	1
PATTERN 5	0	×	1	1	×	×
PATTERN 6	0	×	0	×	×	0

TEST VECTOR
TEST SEQUENCE

1 : HIGH LEVEL SIGNAL
0 : LOW LEVEL SIGNAL
× : DON'T CARE

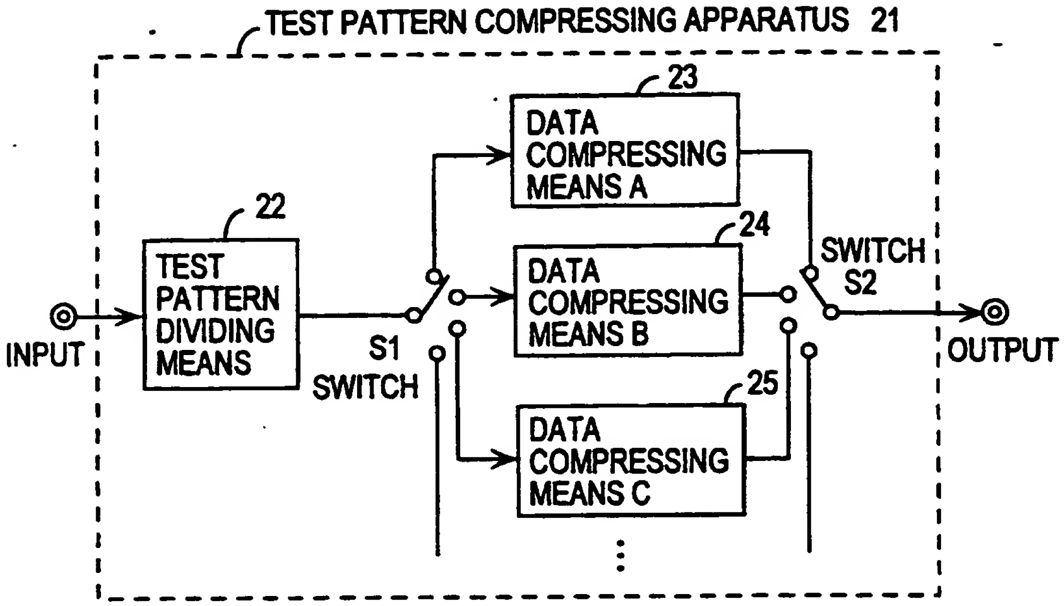

TEST PATTERN COMPRESSING APPARATUS 21
22
TEST PATTERN DIVIDING MEANS
INPUT
S1
SWITCH
23
DATA COMPRESSING MEANS A
24
DATA COMPRESSING MEANS B
25
DATA COMPRESSING MEANS C
SWITCH S2
OUTPUT

FIG. 2

✓

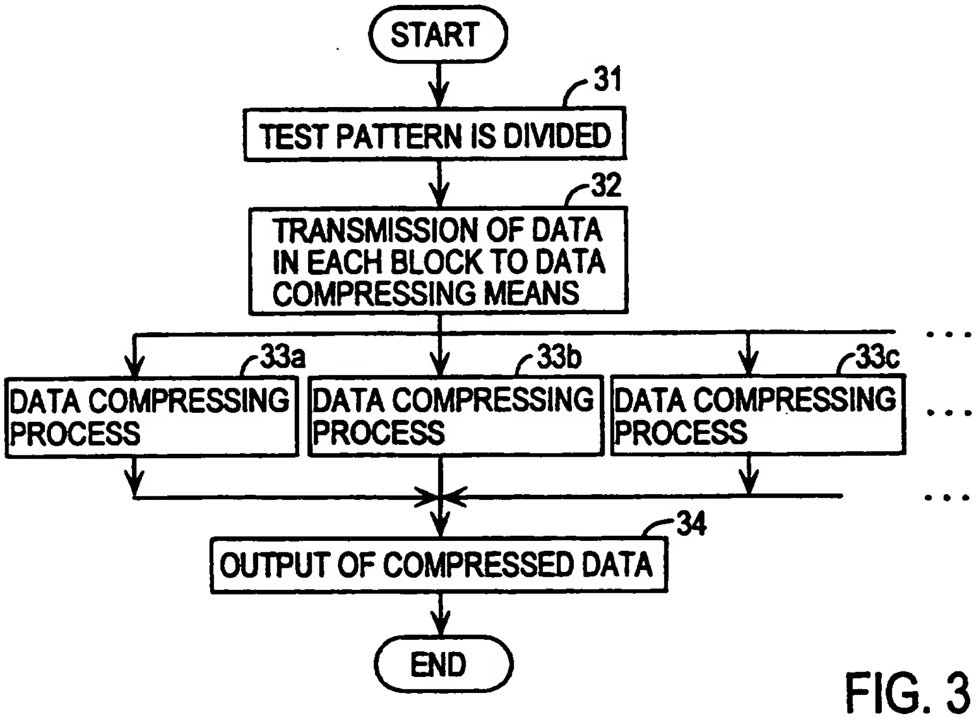

START
31
TEST PATTERN IS DIVIDED
32
TRANSMISSION OF DATA IN EACH BLOCK TO DATA COMPRESSING MEANS
33a
DATA COMPRESSING PROCESS
33b
DATA COMPRESSING PROCESS
33c
DATA COMPRESSING PROCESS
34
OUTPUT OF COMPRESSED DATA
END

FIG. 3

✓

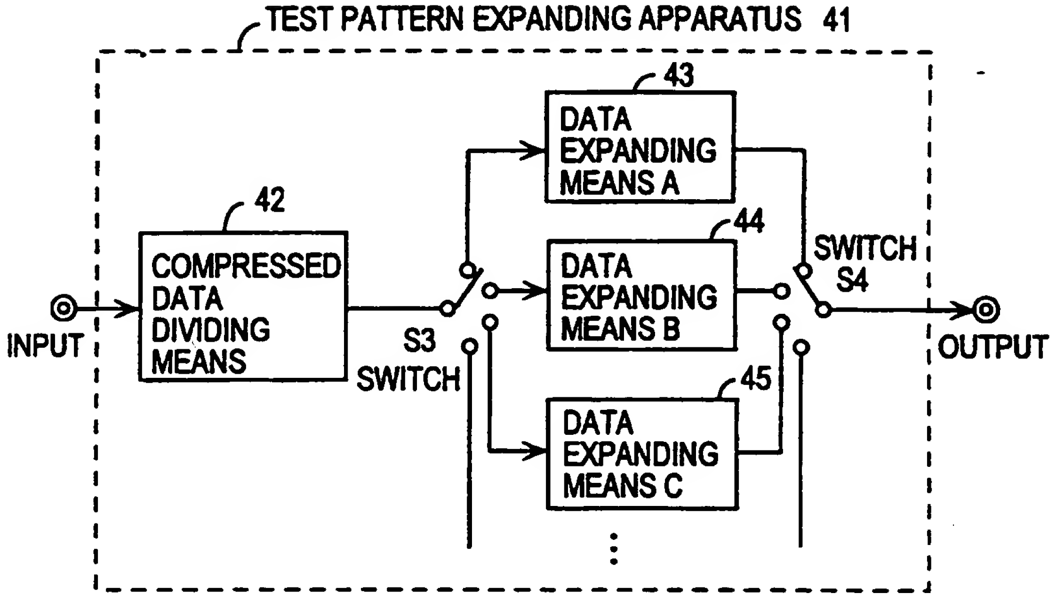

TEST PATTERN EXPANDING APPARATUS 41
43
DATA EXPANDING MEANS A
42
44
COMPRESSED DATA DIVIDING MEANS
DATA EXPANDING MEANS B
SWITCH S4
INPUT
S3 SWITCH
OUTPUT
45
DATA EXPANDING MEANS C

FIG. 4

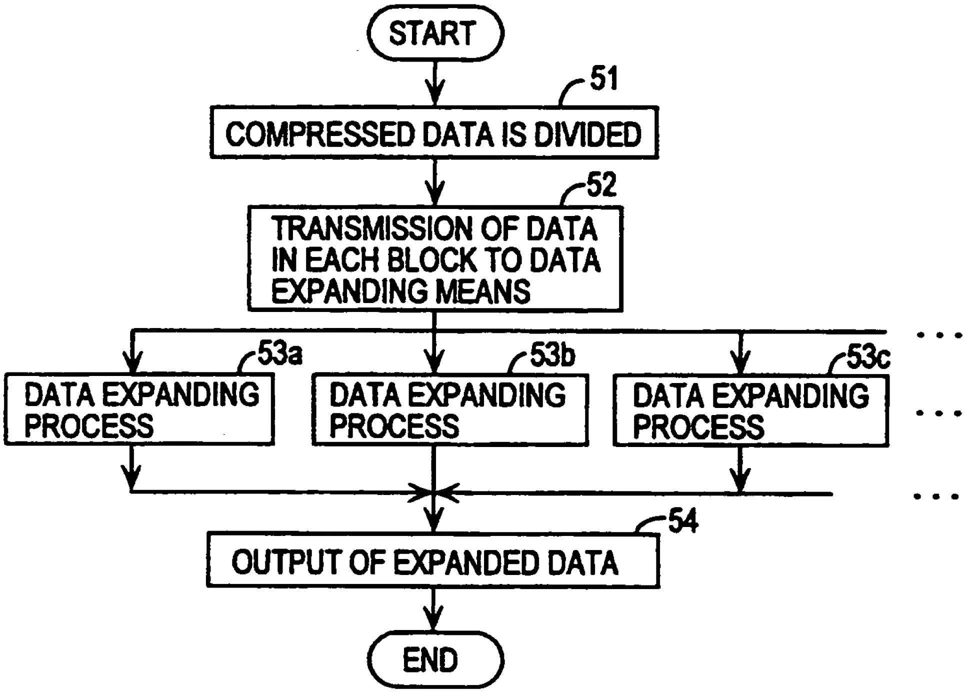

START
51
COMPRESSED DATA IS DIVIDED
52
TRANSMISSION OF DATA IN EACH BLOCK TO DATA EXPANDING MEANS
53a
DATA EXPANDING PROCESS
53b
DATA EXPANDING PROCESS
53c
DATA EXPANDING PROCESS
54
OUTPUT OF EXPANDED DATA
END

FIG. 5

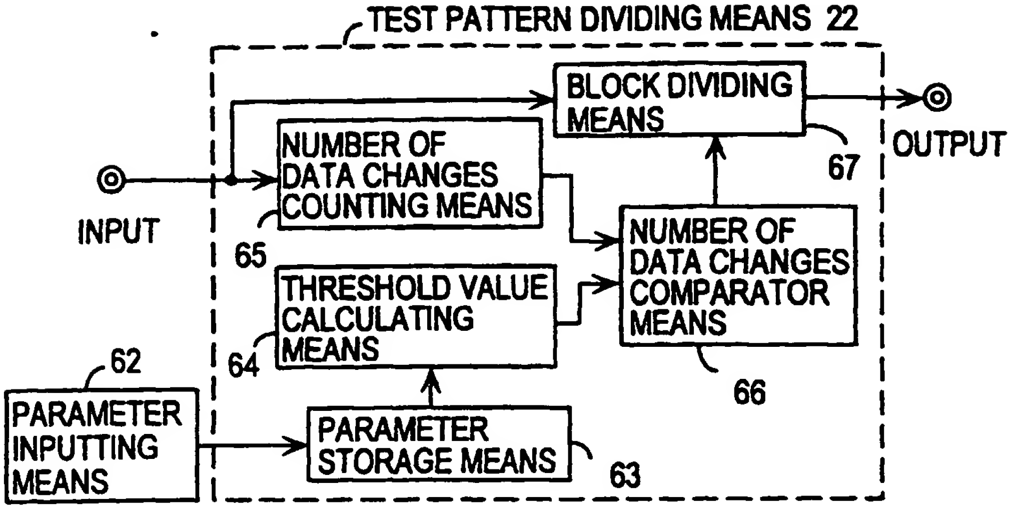

TEST PATTERN DIVIDING MEANS 22
BLOCK DIVIDING MEANS
OUTPUT
67
NUMBER OF DATA CHANGES COUNTING MEANS
INPUT
65
NUMBER OF DATA CHANGES COMPARATOR MEANS
THRESHOLD VALUE CALCULATING MEANS
64
62
PARAMETER INPUTTING MEANS
PARAMETER STORAGE MEANS
63
66

FIG. 6

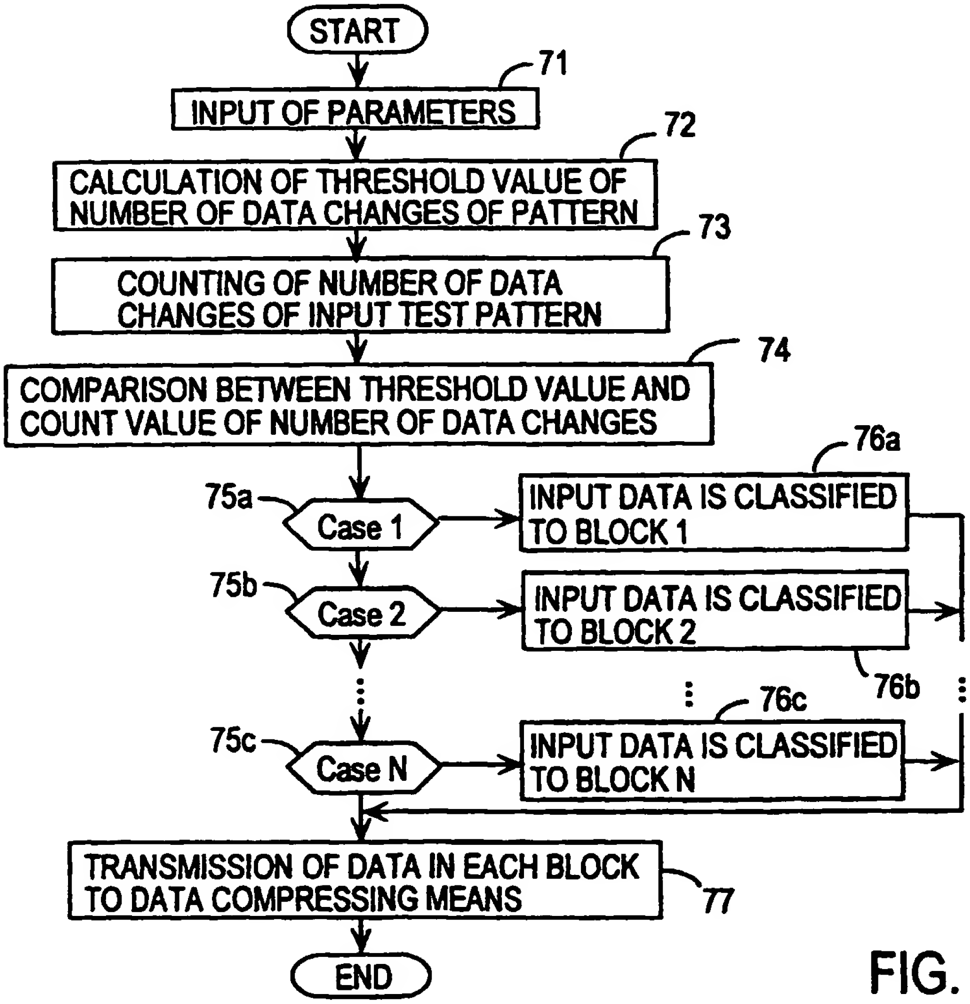

START
71
INPUT OF PARAMETERS
72
CALCULATION OF THRESHOLD VALUE OF NUMBER OF DATA CHANGES OF PATTERN
73
COUNTING OF NUMBER OF DATA CHANGES OF INPUT TEST PATTERN
74
COMPARISON BETWEEN THRESHOLD VALUE AND COUNT VALUE OF NUMBER OF DATA CHANGES
76a
75a
Case 1
INPUT DATA IS CLASSIFIED TO BLOCK 1
75b
Case 2
INPUT DATA IS CLASSIFIED TO BLOCK 2
76b
76c
75c
Case N
INPUT DATA IS CLASSIFIED TO BLOCK N
TRANSMISSION OF DATA IN EACH BLOCK TO DATA COMPRESSING MEANS
77
END

FIG. 7

FIG. 8

(a)

	PIN 1	PIN 2	PIN 3	PIN 4	PIN 5	PIN 6
PATTERN 1	0	0	1	×	×	×
PATTERN 2	0	0	×	1	×	1
PATTERN 3	0	×	×	1	×	×
PATTERN 4	0	×	1	1	×	1
PATTERN 5	0	×	1	1	×	×
PATTERN 6	0	×	0	×	×	0

(b) $\phi_{max} = 2$

(c) $\phi = [\,0 \quad 1 \quad 3 \quad 2 \quad 0 \quad 5\,]$

(d)

BLOCK 1				BLOCK 2	
0	0	×	×	1	×
0	0	1	×	×	1
0	×	1	×	×	×
0	×	1	×	1	1
0	×	1	×	1	×
0	×	×	×	0	0
($\phi \leqq \phi_{max}$)				($\phi > \phi_{max}$)	

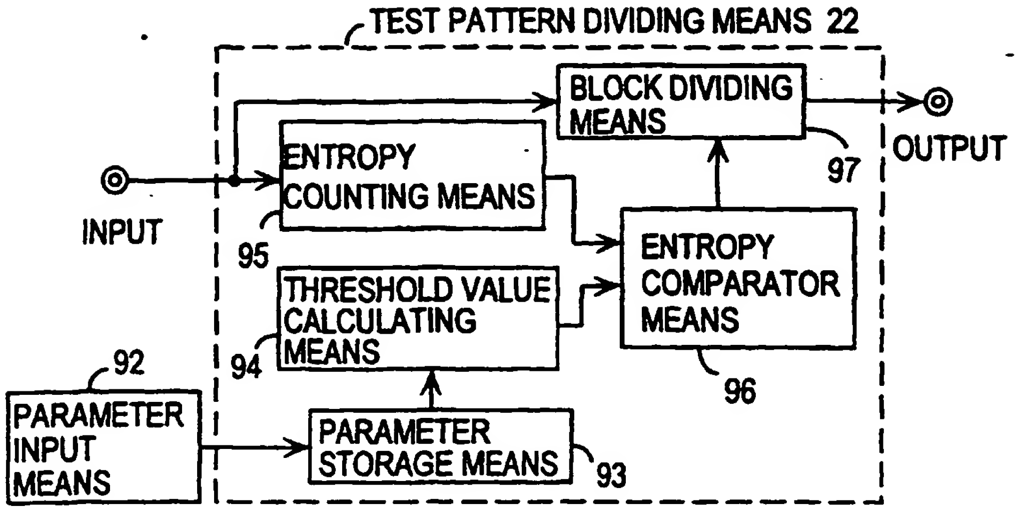

TEST PATTERN DIVIDING MEANS 22
BLOCK DIVIDING MEANS
97
OUTPUT
ENTROPY COUNTING MEANS
95
INPUT
ENTROPY COMPARATOR MEANS
96
THRESHOLD VALUE CALCULATING MEANS
94
92
PARAMETER INPUT MEANS
PARAMETER STORAGE MEANS
93

FIG. 9

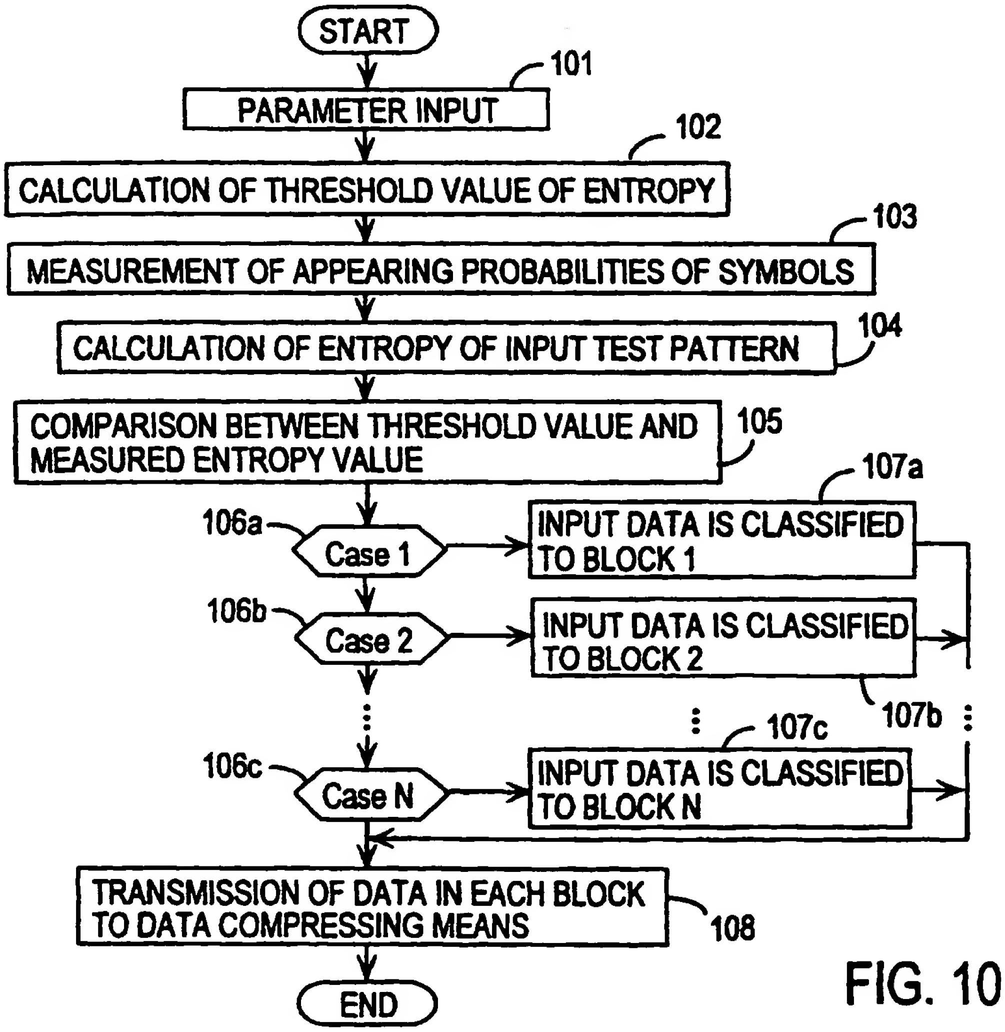

START
101
PARAMETER INPUT
102
CALCULATION OF THRESHOLD VALUE OF ENTROPY
103
MEASUREMENT OF APPEARING PROBABILITIES OF SYMBOLS
104
CALCULATION OF ENTROPY OF INPUT TEST PATTERN
105
COMPARISON BETWEEN THRESHOLD VALUE AND MEASURED ENTROPY VALUE
107a
106a
Case 1
INPUT DATA IS CLASSIFIED TO BLOCK 1
106b
Case 2
INPUT DATA IS CLASSIFIED TO BLOCK 2
107b
107c
106c
Case N
INPUT DATA IS CLASSIFIED TO BLOCK N
TRANSMISSION OF DATA IN EACH BLOCK TO DATA COMPRESSING MEANS
108
END

FIG. 10

✓

FIG. 11

(a)

	PIN 1	PIN 2	PIN 3	PIN 4	PIN 5	PIN 6
PATTERN 1	0	0	1	×	×	×
PATTERN 2	0	0	×	1	×	1
PATTERN 3	0	×	×	1	×	×
PATTERN 4	0	×	1	1	×	1
PATTERN 5	0	×	1	1	×	×
PATTERN 6	0	×	0	×	×	0

(b) $H_{max} = 1.0$

(c) $H = [\,0.0 \;\; 0.92 \;\; 1.46 \;\; 0.92 \;\; 0.0 \;\; 1.46\,]$

(d)

BLOCK 1				BLOCK 2	
0	0	×	×	1	×
0	0	1	×	×	1
0	×	1	×	×	×
0	×	1	×	1	1
0	×	1	×	1	×
0	×	×	×	0	0
$(H \leqq H_{max})$				$(H > H_{max})$	

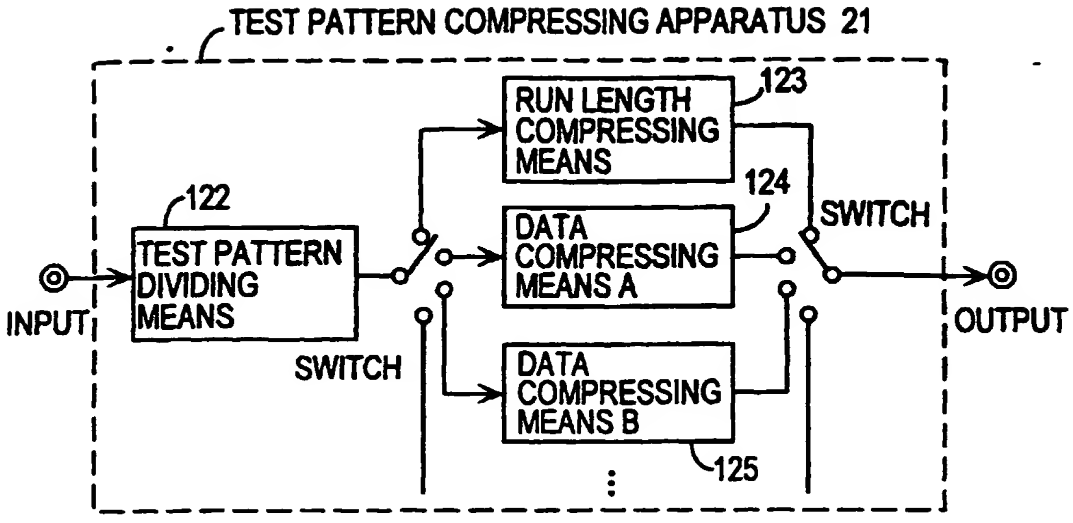

TEST PATTERN COMPRESSING APPARATUS 21
RUN LENGTH COMPRESSING MEANS
123
124
122
TEST PATTERN DIVIDING MEANS
DATA COMPRESSING MEANS A
SWITCH
INPUT
OUTPUT
SWITCH
DATA COMPRESSING MEANS B
125

FIG. 12

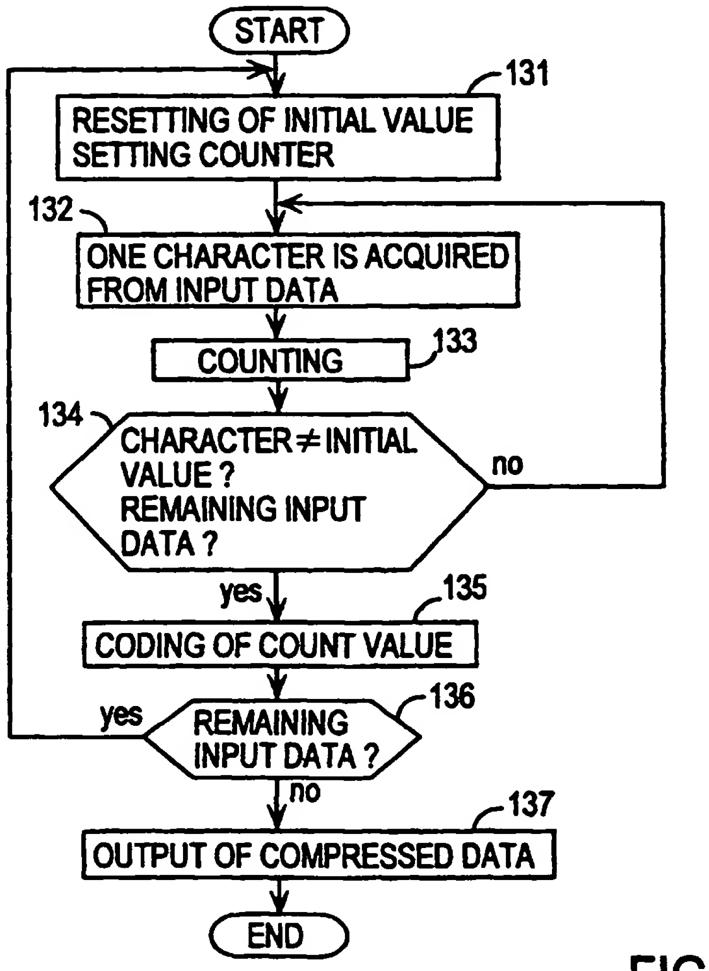

START
131
RESETTING OF INITIAL VALUE SETTING COUNTER
132
ONE CHARACTER IS ACQUIRED FROM INPUT DATA
133
COUNTING
134
CHARACTER≠INITIAL VALUE ? REMAINING INPUT DATA ?
no
yes
135
CODING OF COUNT VALUE
136
yes
REMAINING INPUT DATA ?
no
137
OUTPUT OF COMPRESSED DATA
END

FIG. 13

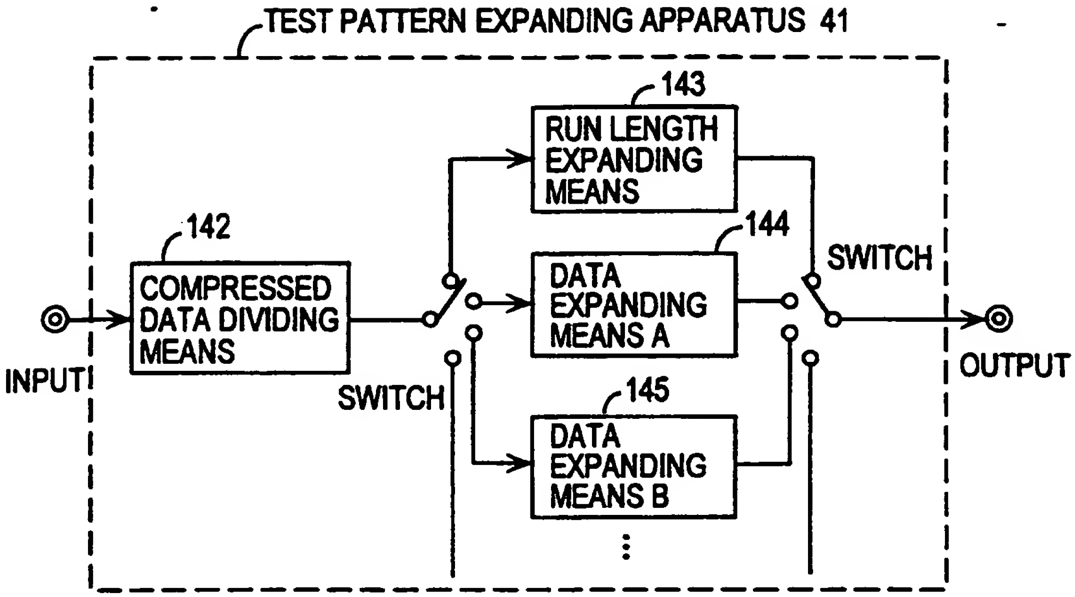

TEST PATTERN EXPANDING APPARATUS 41
142
COMPRESSED DATA DIVIDING MEANS
INPUT
SWITCH
143
RUN LENGTH EXPANDING MEANS
144
DATA EXPANDING MEANS A
145
DATA EXPANDING MEANS B
SWITCH
OUTPUT

FIG. 14

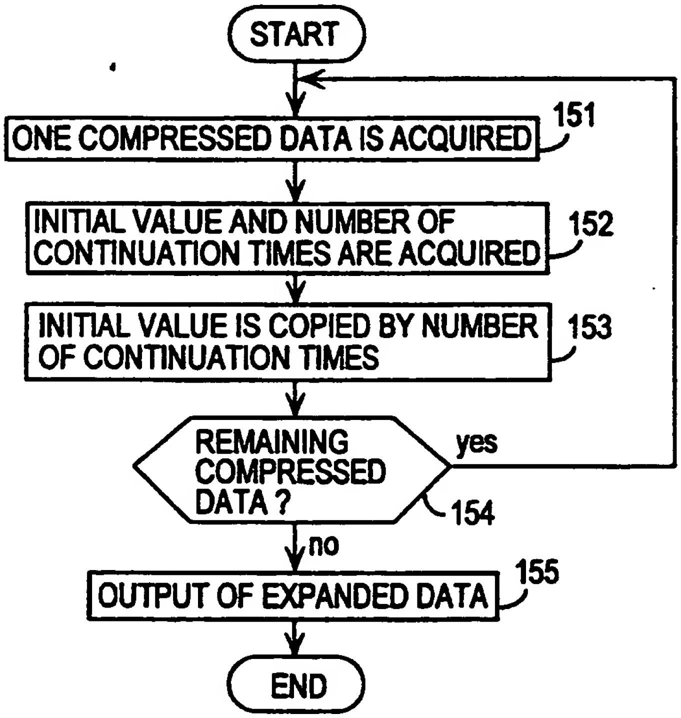

START
ONE COMPRESSED DATA IS ACQUIRED
151
INITIAL VALUE AND NUMBER OF CONTINUATION TIMES ARE ACQUIRED
152
INITIAL VALUE IS COPIED BY NUMBER OF CONTINUATION TIMES
153
REMAINING COMPRESSED DATA ?
yes
154
no
OUTPUT OF EXPANDED DATA
155
END

FIG. 15

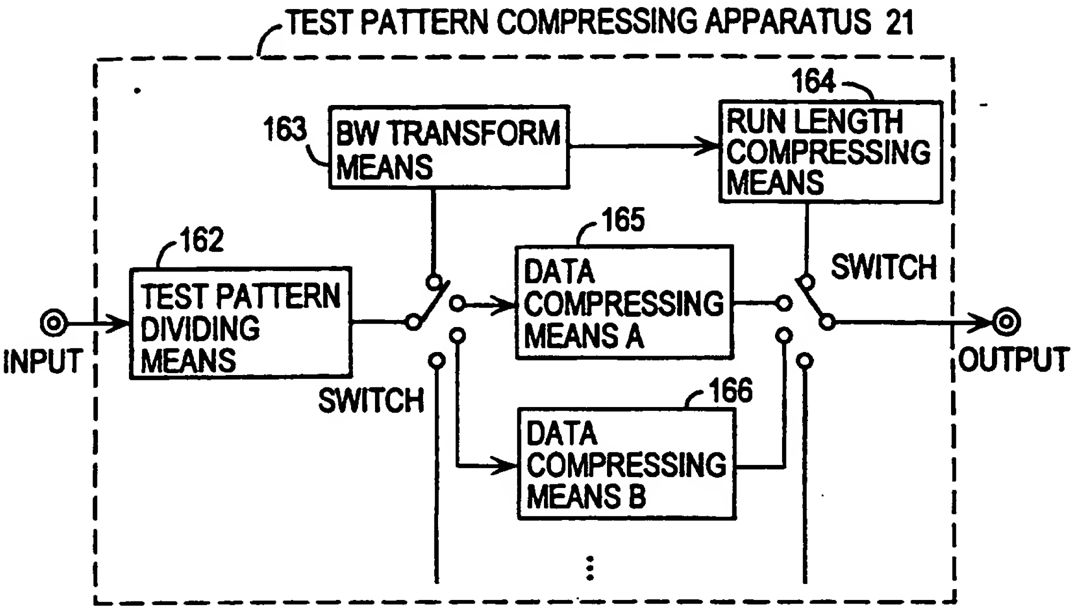

TEST PATTERN COMPRESSING APPARATUS 21
163
BW TRANSFORM MEANS
164
RUN LENGTH COMPRESSING MEANS
162
TEST PATTERN DIVIDING MEANS
165
DATA COMPRESSING MEANS A
SWITCH
INPUT
SWITCH
OUTPUT
166
DATA COMPRESSING MEANS B

FIG. 16

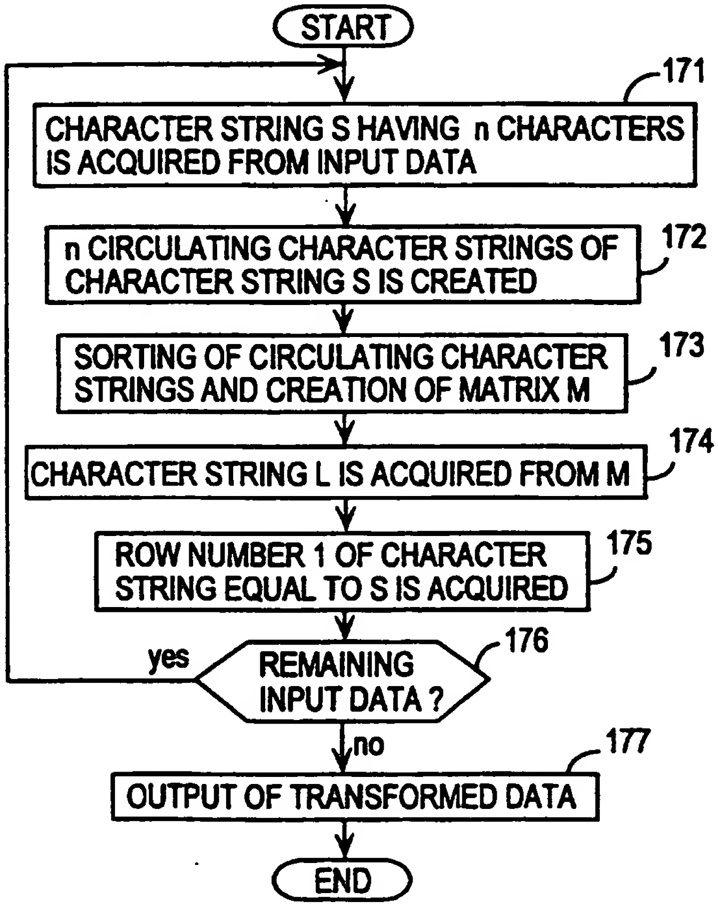

START
171
CHARACTER STRING S HAVING n CHARACTERS IS ACQUIRED FROM INPUT DATA
172
n CIRCULATING CHARACTER STRINGS OF CHARACTER STRING S IS CREATED
173
SORTING OF CIRCULATING CHARACTER STRINGS AND CREATION OF MATRIX M
174
CHARACTER STRING L IS ACQUIRED FROM M
175
ROW NUMBER 1 OF CHARACTER STRING EQUAL TO S IS ACQUIRED
176
yes
REMAINING INPUT DATA ?
no
177
OUTPUT OF TRANSFORMED DATA
END

FIG. 17

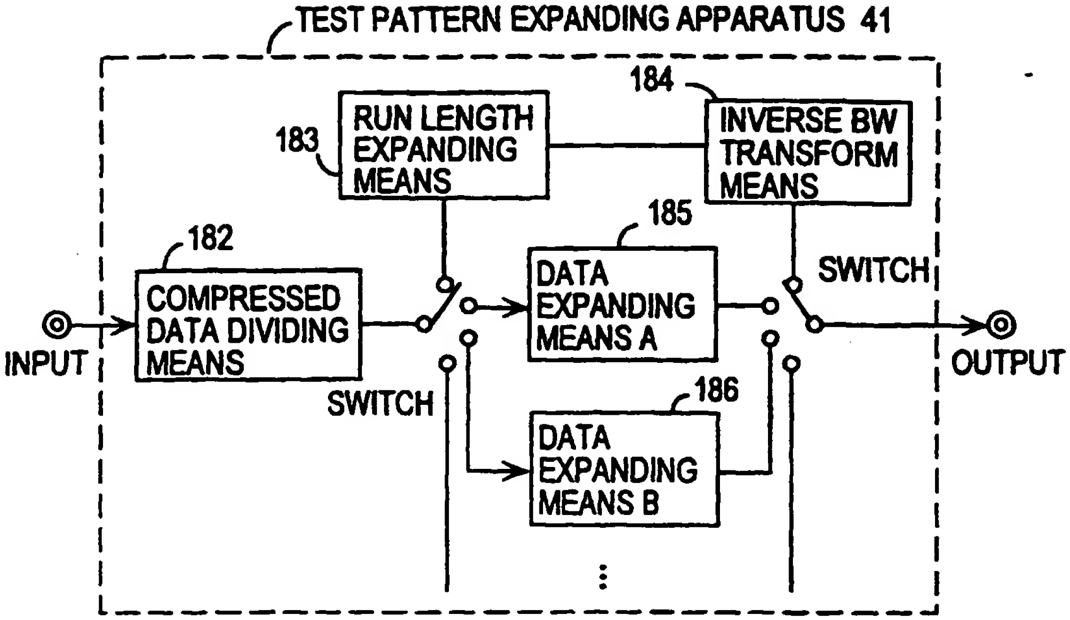

TEST PATTERN EXPANDING APPARATUS 41
183
RUN LENGTH EXPANDING MEANS
184
INVERSE BW TRANSFORM MEANS
182
COMPRESSED DATA DIVIDING MEANS
185
DATA EXPANDING MEANS A
SWITCH
INPUT
SWITCH
OUTPUT
186
DATA EXPANDING MEANS B

FIG. 18

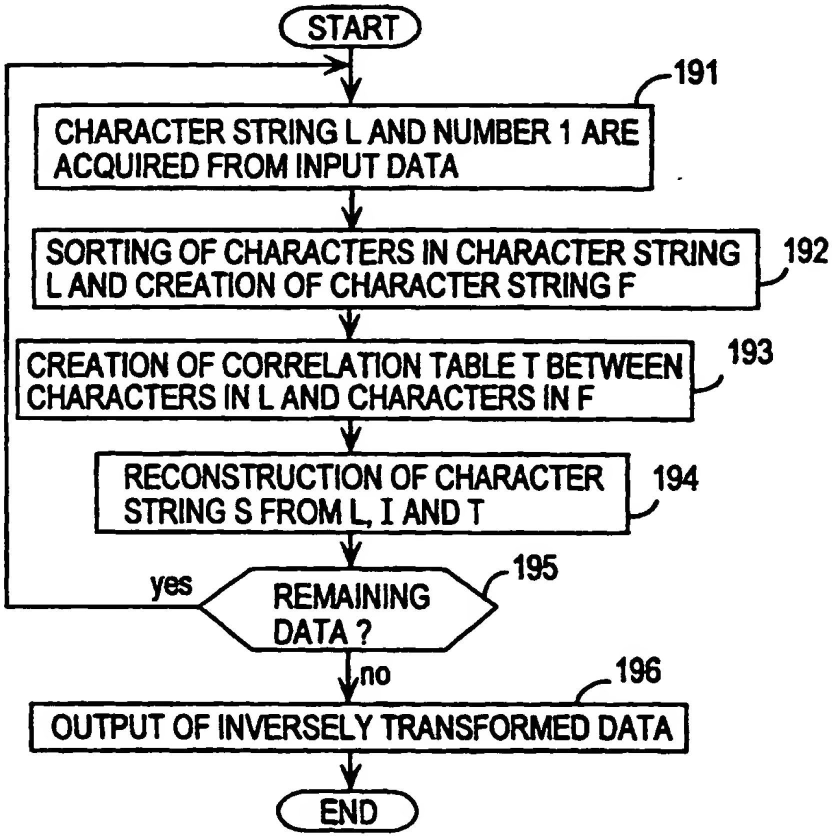

START
191
CHARACTER STRING L AND NUMBER 1 ARE ACQUIRED FROM INPUT DATA
192
SORTING OF CHARACTERS IN CHARACTER STRING L AND CREATION OF CHARACTER STRING F
193
CREATION OF CORRELATION TABLE T BETWEEN CHARACTERS IN L AND CHARACTERS IN F
194
RECONSTRUCTION OF CHARACTER STRING S FROM L, I AND T
195
yes
REMAINING DATA ?
no
196
OUTPUT OF INVERSELY TRANSFORMED DATA
END

FIG. 19

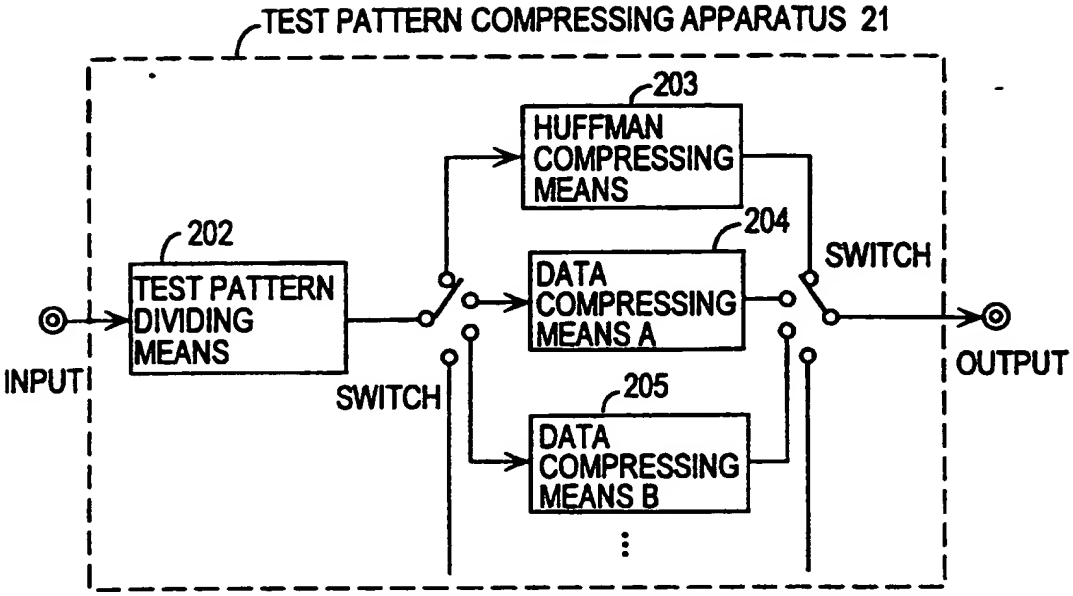

TEST PATTERN COMPRESSING APPARATUS 21
203
HUFFMAN COMPRESSING MEANS
202
TEST PATTERN DIVIDING MEANS
204
DATA COMPRESSING MEANS A
SWITCH
INPUT
SWITCH
205
DATA COMPRESSING MEANS B
OUTPUT

FIG. 20

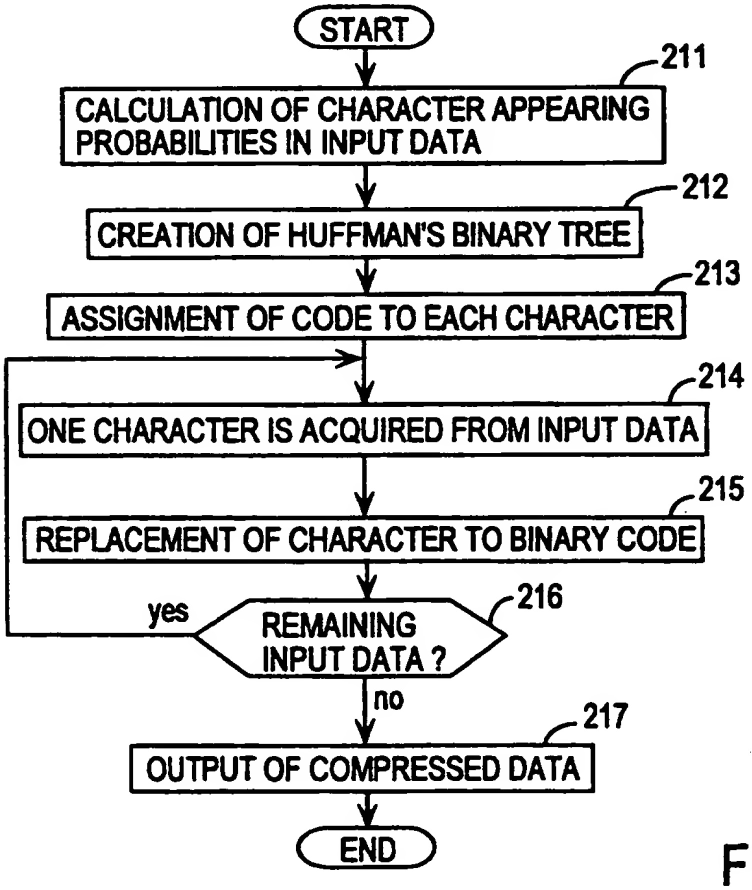

START
211
CALCULATION OF CHARACTER APPEARING PROBABILITIES IN INPUT DATA
212
CREATION OF HUFFMAN'S BINARY TREE
213
ASSIGNMENT OF CODE TO EACH CHARACTER
214
ONE CHARACTER IS ACQUIRED FROM INPUT DATA
215
REPLACEMENT OF CHARACTER TO BINARY CODE
216
yes
REMAINING INPUT DATA ?
no
217
OUTPUT OF COMPRESSED DATA
END

FIG. 21

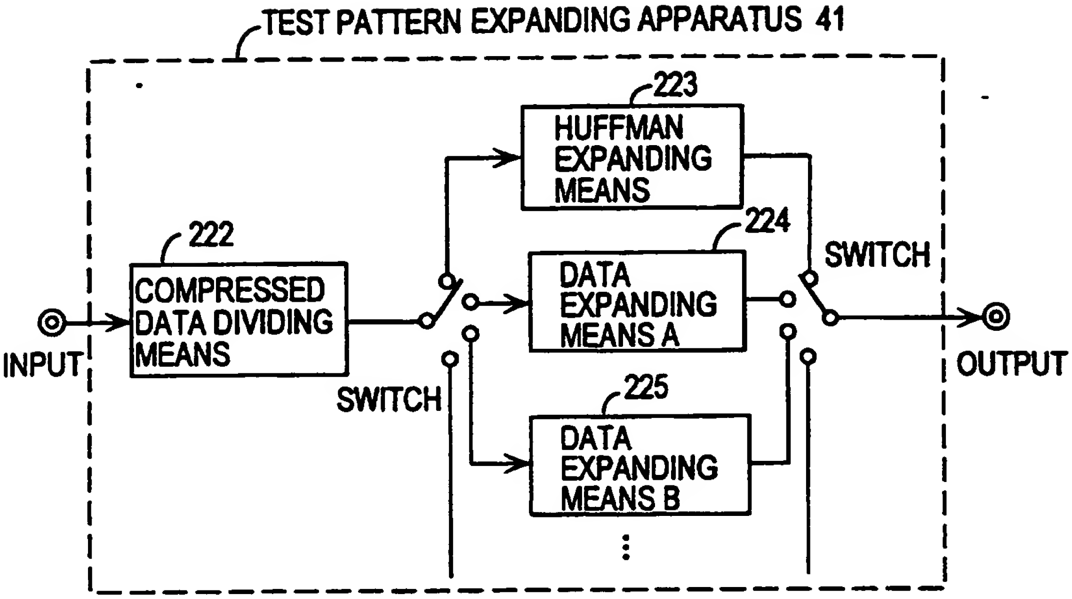

TEST PATTERN EXPANDING APPARATUS 41
223
HUFFMAN EXPANDING MEANS
224
DATA EXPANDING MEANS A
225
DATA EXPANDING MEANS B
222
COMPRESSED DATA DIVIDING MEANS
INPUT
SWITCH
SWITCH
OUTPUT

FIG. 22

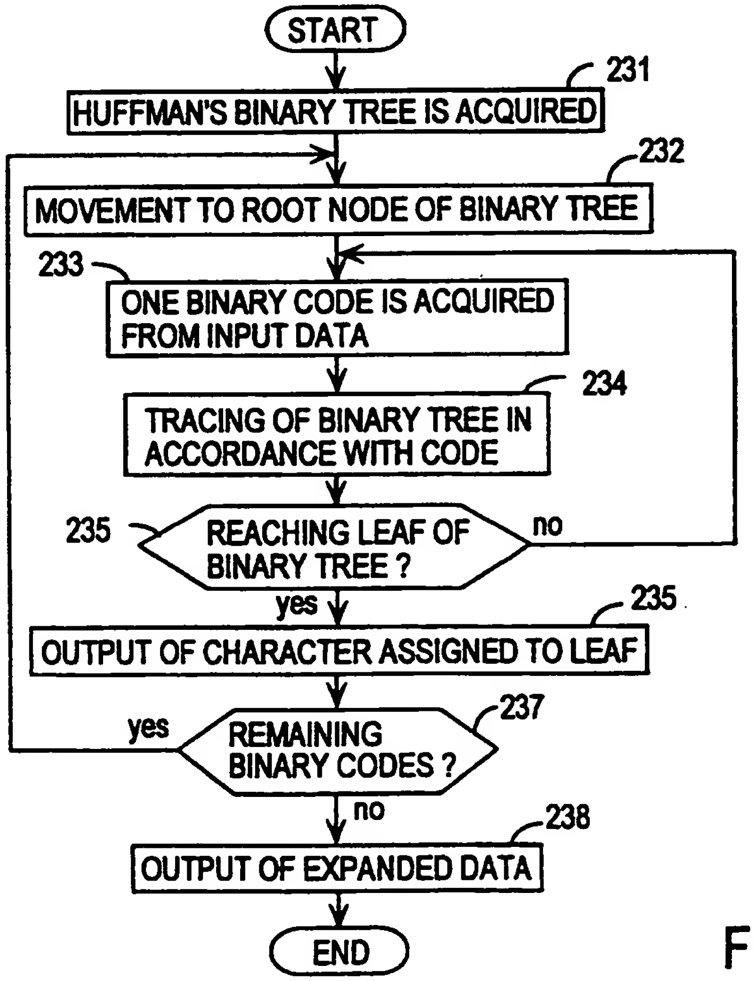

START
231
HUFFMAN'S BINARY TREE IS ACQUIRED
232
MOVEMENT TO ROOT NODE OF BINARY TREE
233
ONE BINARY CODE IS ACQUIRED FROM INPUT DATA
234
TRACING OF BINARY TREE IN ACCORDANCE WITH CODE
235
REACHING LEAF OF BINARY TREE ?
no
yes
235
OUTPUT OF CHARACTER ASSIGNED TO LEAF
237
REMAINING BINARY CODES ?
yes
no
238
OUTPUT OF EXPANDED DATA
END

FIG. 23

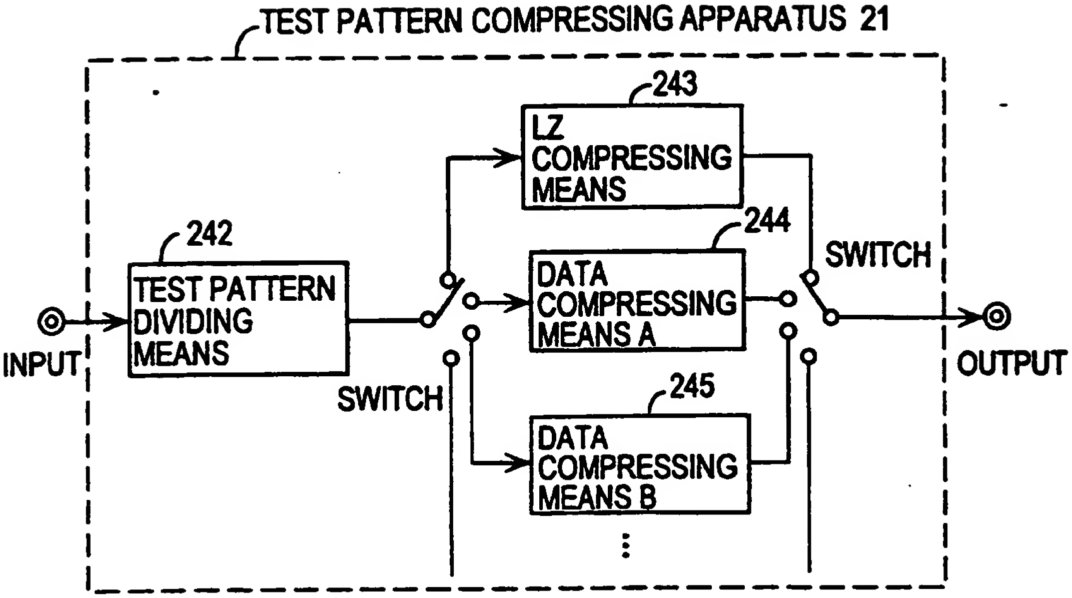

TEST PATTERN COMPRESSING APPARATUS 21
243
LZ COMPRESSING MEANS
242
TEST PATTERN DIVIDING MEANS
244
DATA COMPRESSING MEANS A
SWITCH
INPUT
OUTPUT
SWITCH
245
DATA COMPRESSING MEANS B

FIG. 24

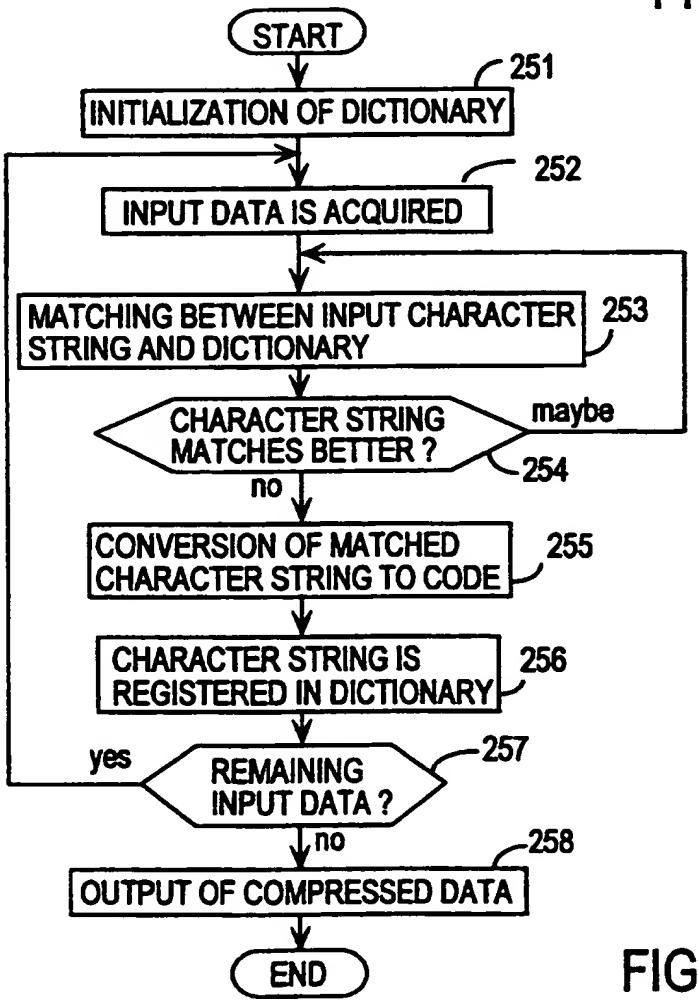

START
251
INITIALIZATION OF DICTIONARY
252
INPUT DATA IS ACQUIRED
MATCHING BETWEEN INPUT CHARACTER STRING AND DICTIONARY
253
CHARACTER STRING MATCHES BETTER ?
maybe
254
no
CONVERSION OF MATCHED CHARACTER STRING TO CODE
255
CHARACTER STRING IS REGISTERED IN DICTIONARY
256
yes
REMAINING INPUT DATA ?
257
no
258
OUTPUT OF COMPRESSED DATA
END

FIG. 25

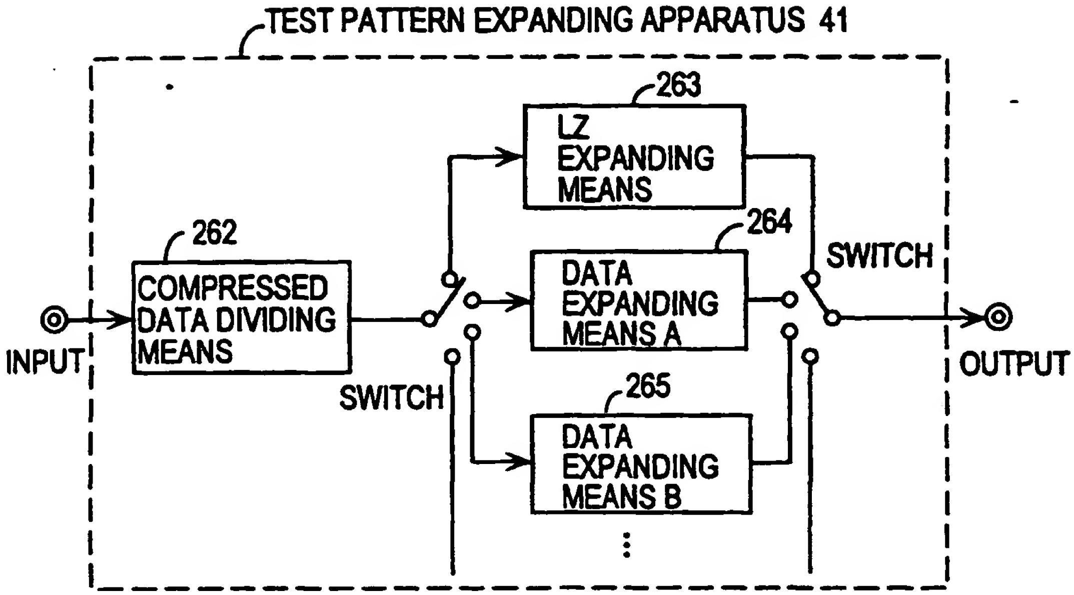

TEST PATTERN EXPANDING APPARATUS 41
263
LZ EXPANDING MEANS
262
COMPRESSED DATA DIVIDING MEANS
264
DATA EXPANDING MEANS A
SWITCH
INPUT
OUTPUT
SWITCH
265
DATA EXPANDING MEANS B

FIG. 26

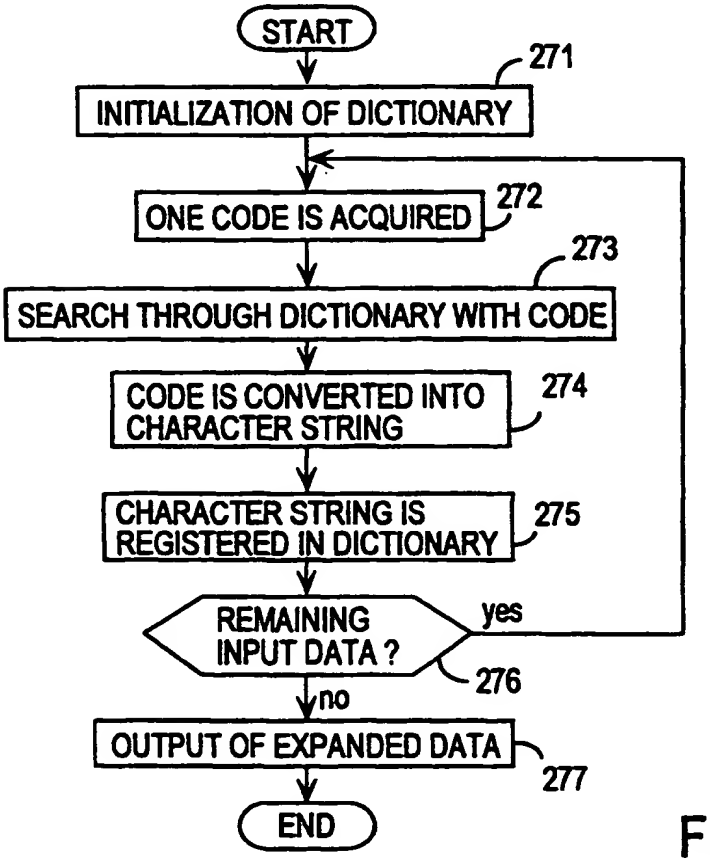

START
271
INITIALIZATION OF DICTIONARY
ONE CODE IS ACQUIRED
272
273
SEARCH THROUGH DICTIONARY WITH CODE
CODE IS CONVERTED INTO CHARACTER STRING
274
CHARACTER STRING IS REGISTERED IN DICTIONARY
275
REMAINING INPUT DATA ?
yes
276
no
OUTPUT OF EXPANDED DATA
277
END

FIG. 27

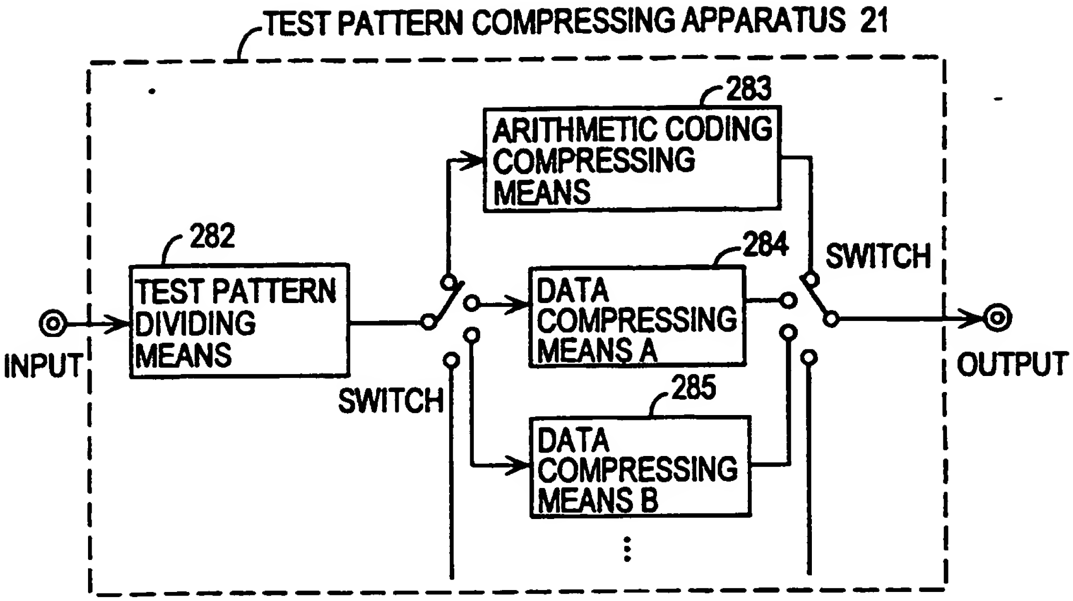

TEST PATTERN COMPRESSING APPARATUS 21
283
ARITHMETIC CODING COMPRESSING MEANS
282
TEST PATTERN DIVIDING MEANS
284
DATA COMPRESSING MEANS A
SWITCH
INPUT
OUTPUT
SWITCH
285
DATA COMPRESSING MEANS B

FIG. 28

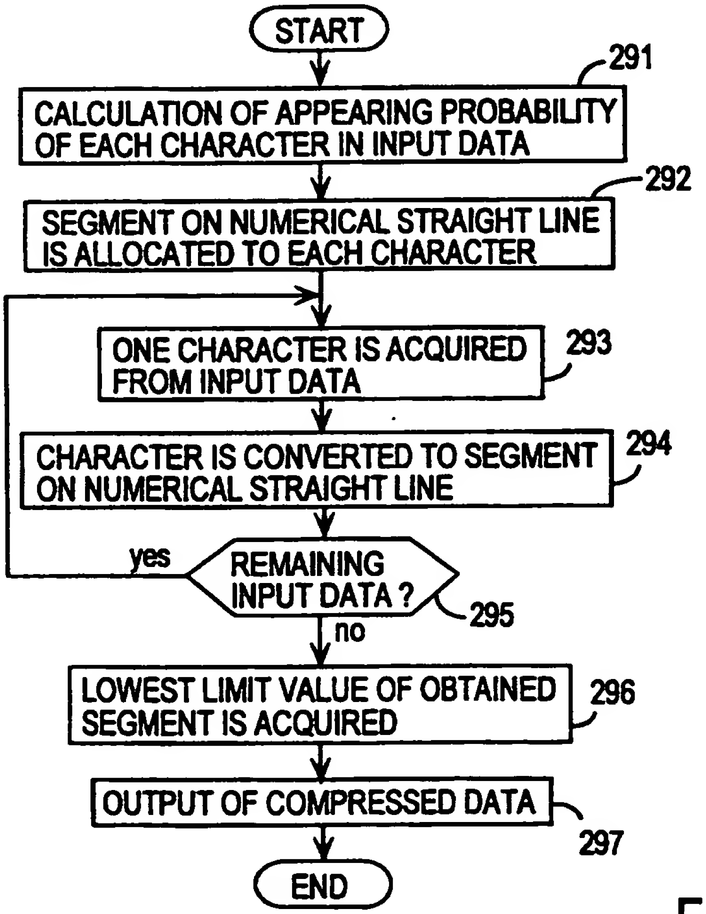

START
291
CALCULATION OF APPEARING PROBABILITY OF EACH CHARACTER IN INPUT DATA
292
SEGMENT ON NUMERICAL STRAIGHT LINE IS ALLOCATED TO EACH CHARACTER
ONE CHARACTER IS ACQUIRED FROM INPUT DATA
293
CHARACTER IS CONVERTED TO SEGMENT ON NUMERICAL STRAIGHT LINE
294
yes
REMAINING INPUT DATA ?
295
no
LOWEST LIMIT VALUE OF OBTAINED SEGMENT IS ACQUIRED
296
OUTPUT OF COMPRESSED DATA
297
END

FIG.29

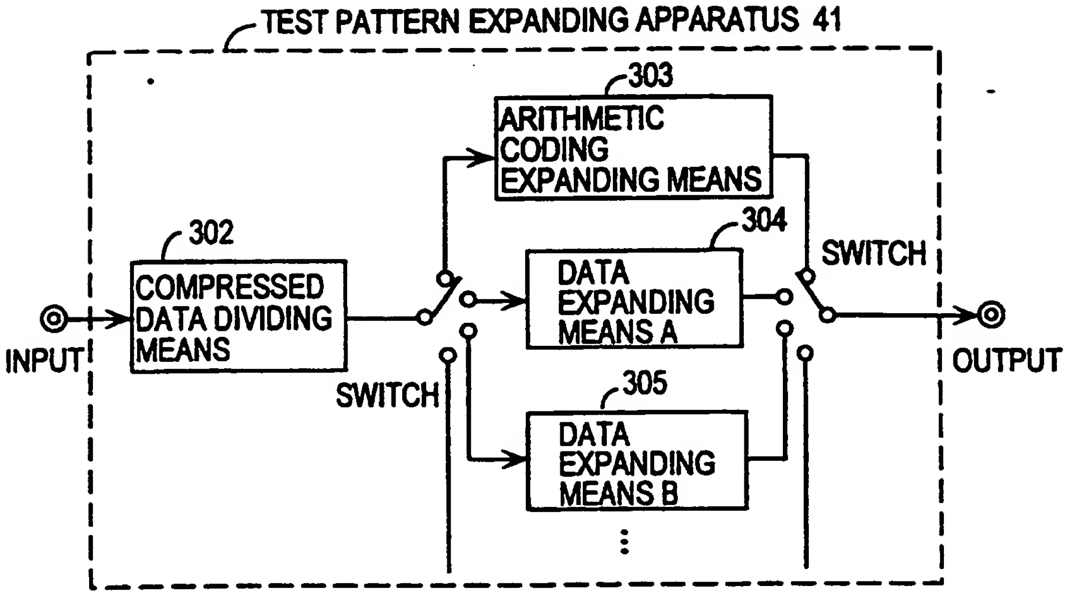

TEST PATTERN EXPANDING APPARATUS 41
303
ARITHMETIC CODING EXPANDING MEANS
302
COMPRESSED DATA DIVIDING MEANS
304
DATA EXPANDING MEANS A
SWITCH
INPUT
OUTPUT
SWITCH
305
DATA EXPANDING MEANS B

FIG. 30

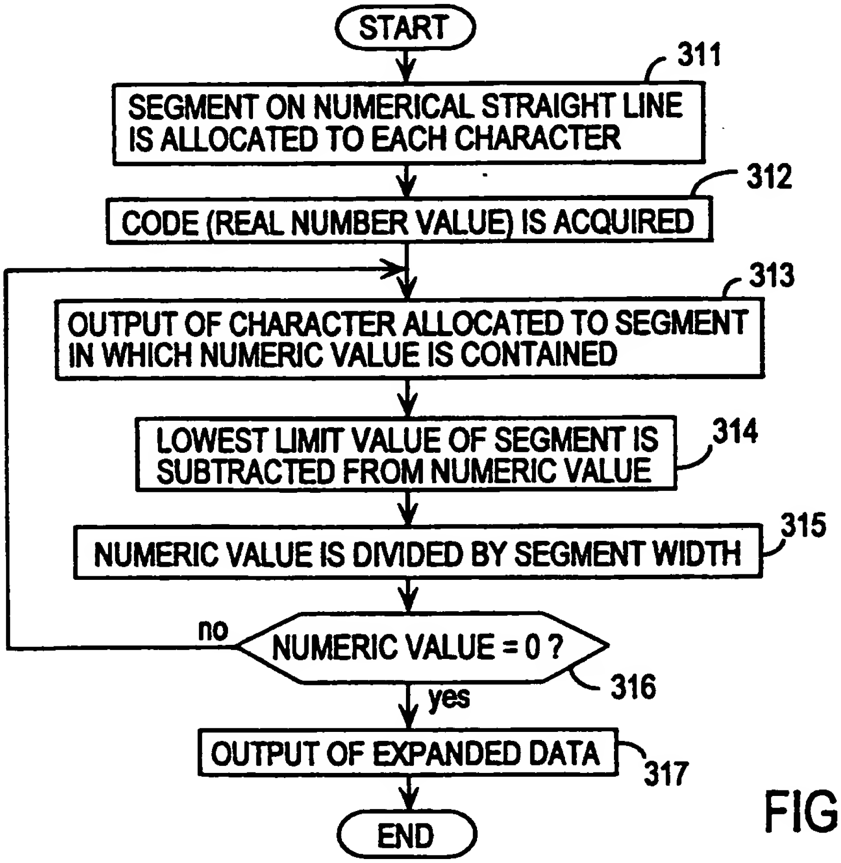

START
311
SEGMENT ON NUMERICAL STRAIGHT LINE IS ALLOCATED TO EACH CHARACTER
312
CODE (REAL NUMBER VALUE) IS ACQUIRED
313
OUTPUT OF CHARACTER ALLOCATED TO SEGMENT IN WHICH NUMERIC VALUE IS CONTAINED
314
LOWEST LIMIT VALUE OF SEGMENT IS SUBTRACTED FROM NUMERIC VALUE
315
NUMERIC VALUE IS DIVIDED BY SEGMENT WIDTH
no
NUMERIC VALUE = 0 ?
316
yes
OUTPUT OF EXPANDED DATA
317
END

FIG. 31

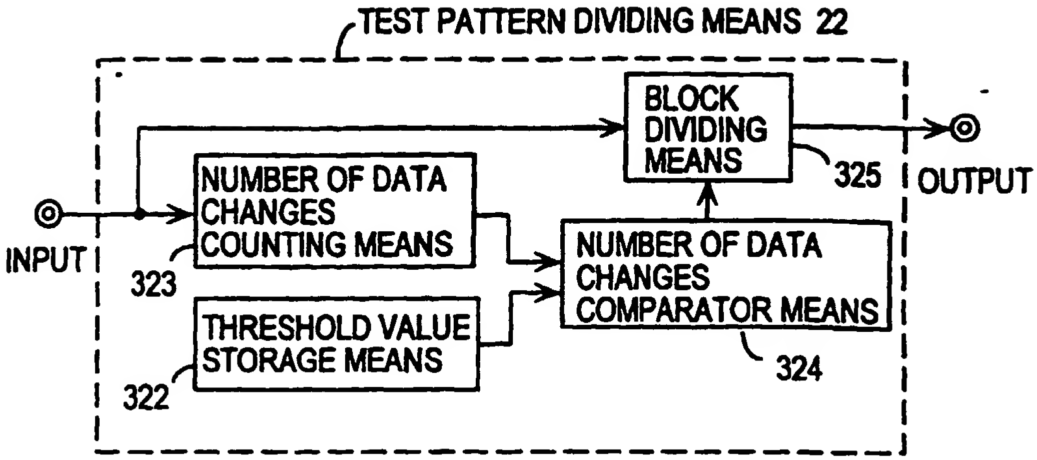

TEST PATTERN DIVIDING MEANS 22
INPUT
NUMBER OF DATA CHANGES COUNTING MEANS
323
THRESHOLD VALUE STORAGE MEANS
322
BLOCK DIVIDING MEANS
325
OUTPUT
NUMBER OF DATA CHANGES COMPARATOR MEANS
324

FIG. 32

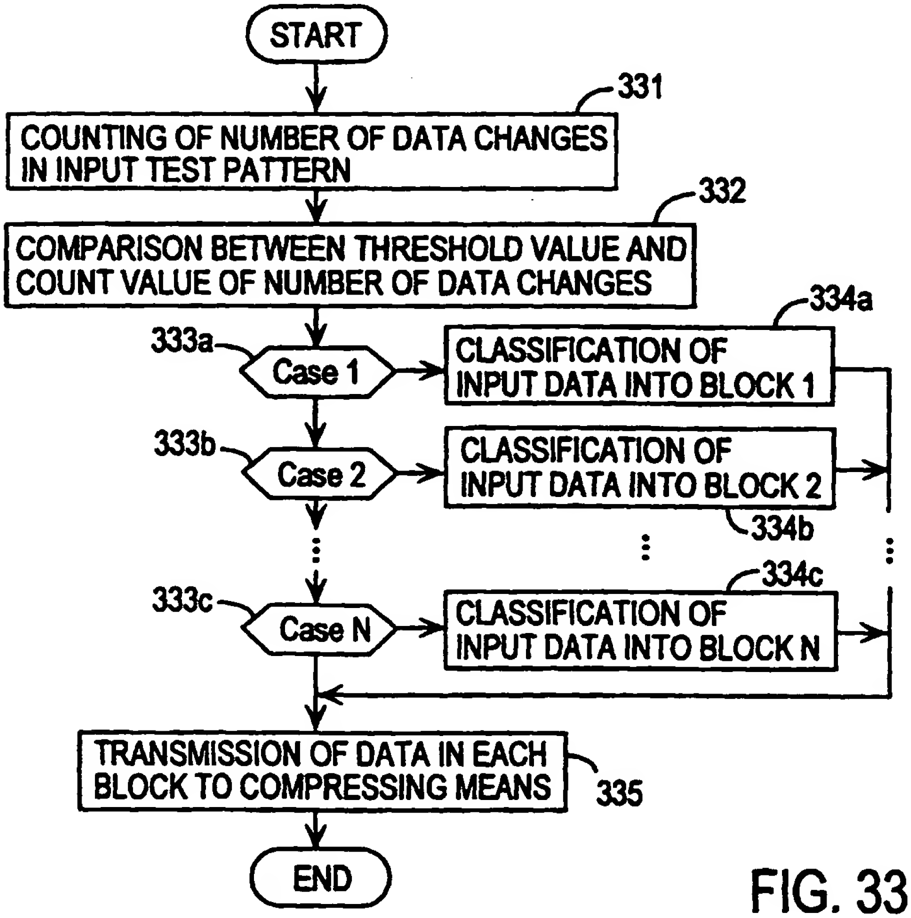

START
331
COUNTING OF NUMBER OF DATA CHANGES IN INPUT TEST PATTERN
332
COMPARISON BETWEEN THRESHOLD VALUE AND COUNT VALUE OF NUMBER OF DATA CHANGES
333a
Case 1
334a
CLASSIFICATION OF INPUT DATA INTO BLOCK 1
333b
Case 2
CLASSIFICATION OF INPUT DATA INTO BLOCK 2
334b
333c
Case N
334c
CLASSIFICATION OF INPUT DATA INTO BLOCK N
TRANSMISSION OF DATA IN EACH BLOCK TO COMPRESSING MEANS
335
END

FIG. 33

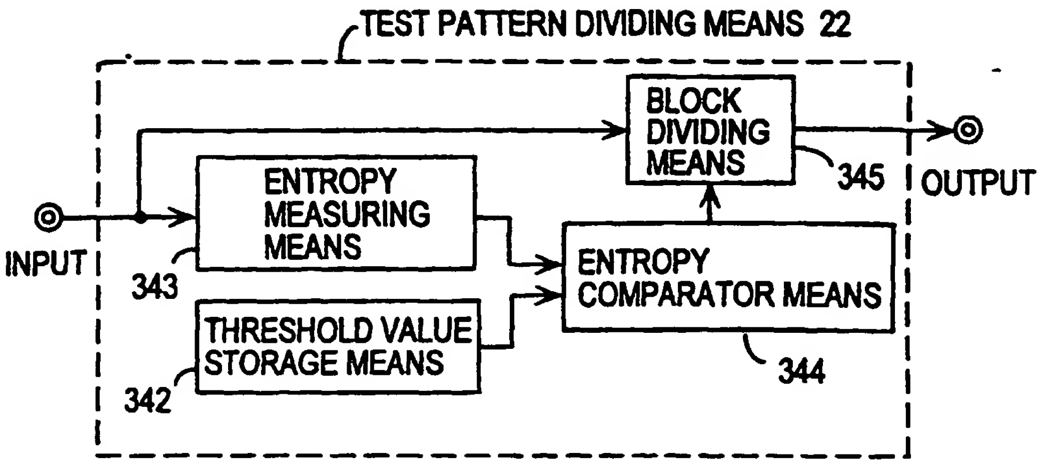

TEST PATTERN DIVIDING MEANS 22
BLOCK DIVIDING MEANS
345
OUTPUT
ENTROPY MEASURING MEANS
INPUT
343
ENTROPY COMPARATOR MEANS
THRESHOLD VALUE STORAGE MEANS
344
342

FIG. 34

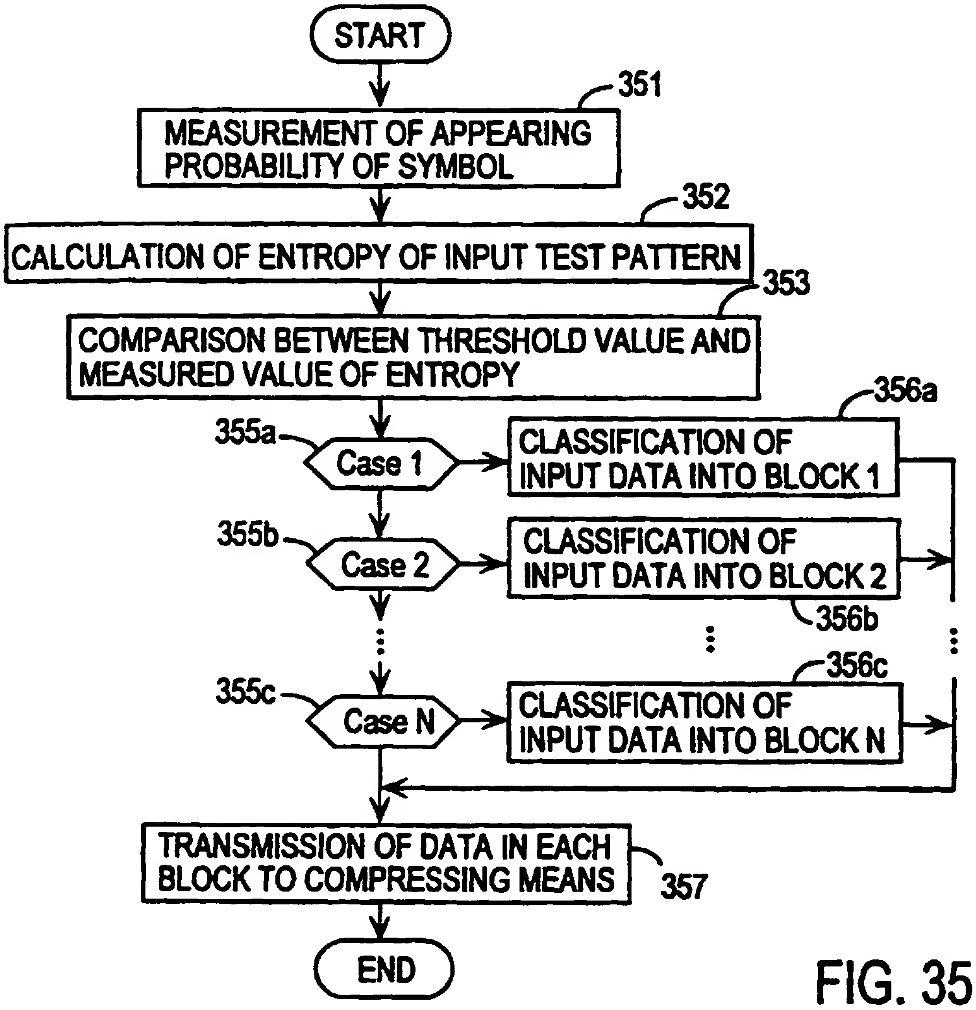

START
351
MEASUREMENT OF APPEARING PROBABILITY OF SYMBOL
352
CALCULATION OF ENTROPY OF INPUT TEST PATTERN
353
COMPARISON BETWEEN THRESHOLD VALUE AND MEASURED VALUE OF ENTROPY
356a
355a
Case 1
CLASSIFICATION OF INPUT DATA INTO BLOCK 1
355b
Case 2
CLASSIFICATION OF INPUT DATA INTO BLOCK 2
356b
356c
355c
Case N
CLASSIFICATION OF INPUT DATA INTO BLOCK N
TRANSMISSION OF DATA IN EACH BLOCK TO COMPRESSING MEANS
357
END

FIG. 35

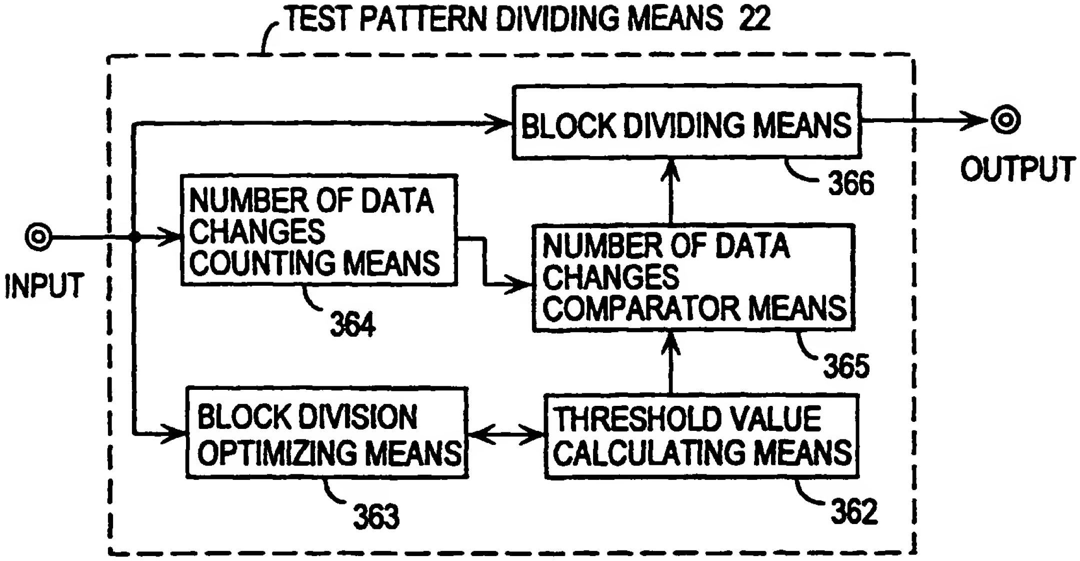

TEST PATTERN DIVIDING MEANS 22
INPUT
OUTPUT
BLOCK DIVIDING MEANS
366
NUMBER OF DATA CHANGES COUNTING MEANS
364
NUMBER OF DATA CHANGES COMPARATOR MEANS
365
BLOCK DIVISION OPTIMIZING MEANS
363
THRESHOLD VALUE CALCULATING MEANS
362

FIG. 36

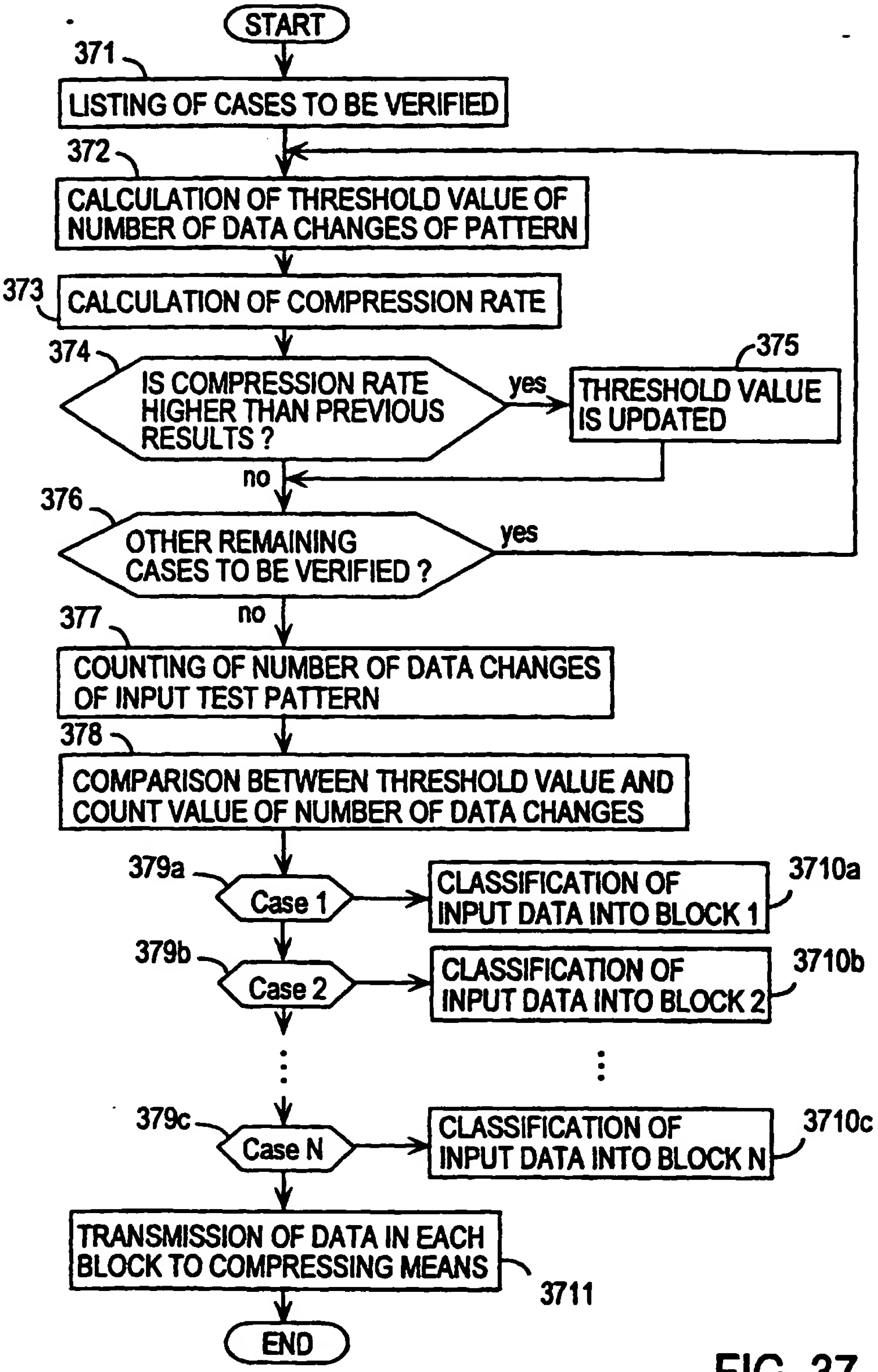

START
371
LISTING OF CASES TO BE VERIFIED
372
CALCULATION OF THRESHOLD VALUE OF NUMBER OF DATA CHANGES OF PATTERN
373
CALCULATION OF COMPRESSION RATE
374
IS COMPRESSION RATE HIGHER THAN PREVIOUS RESULTS ?
yes
375
THRESHOLD VALUE IS UPDATED
no
376
OTHER REMAINING CASES TO BE VERIFIED ?
yes
377
no
COUNTING OF NUMBER OF DATA CHANGES OF INPUT TEST PATTERN
378
COMPARISON BETWEEN THRESHOLD VALUE AND COUNT VALUE OF NUMBER OF DATA CHANGES
379a
Case 1
CLASSIFICATION OF INPUT DATA INTO BLOCK 1
3710a
379b
Case 2
CLASSIFICATION OF INPUT DATA INTO BLOCK 2
3710b
379c
Case N
CLASSIFICATION OF INPUT DATA INTO BLOCK N
3710c
TRANSMISSION OF DATA IN EACH BLOCK TO COMPRESSING MEANS
3711
END

FIG. 37

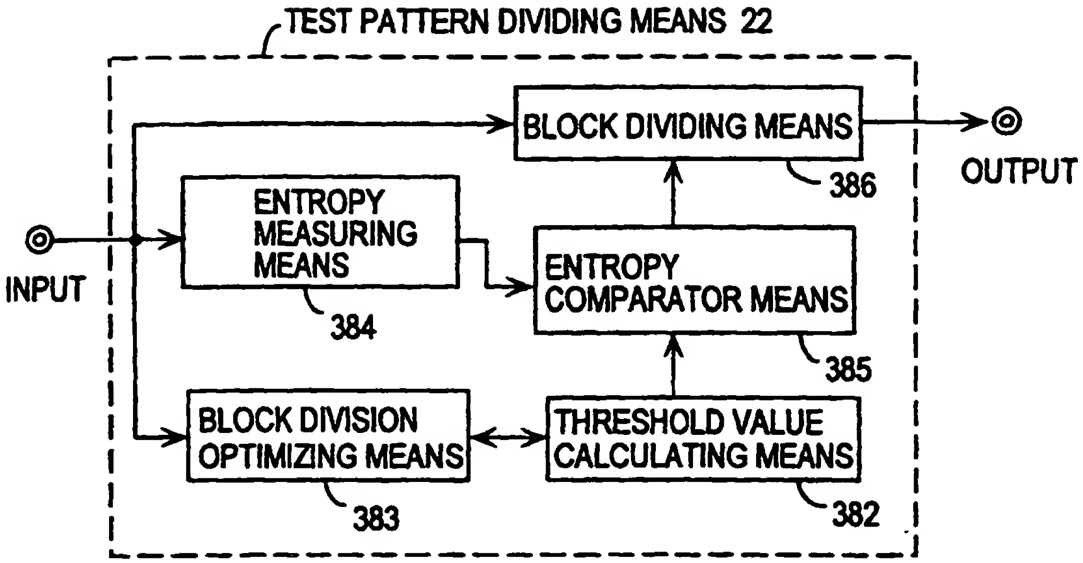

TEST PATTERN DIVIDING MEANS 22
BLOCK DIVIDING MEANS
386
OUTPUT
ENTROPY MEASURING MEANS
384
INPUT
ENTROPY COMPARATOR MEANS
385
BLOCK DIVISION OPTIMIZING MEANS
383
THRESHOLD VALUE CALCULATING MEANS
382

FIG. 38

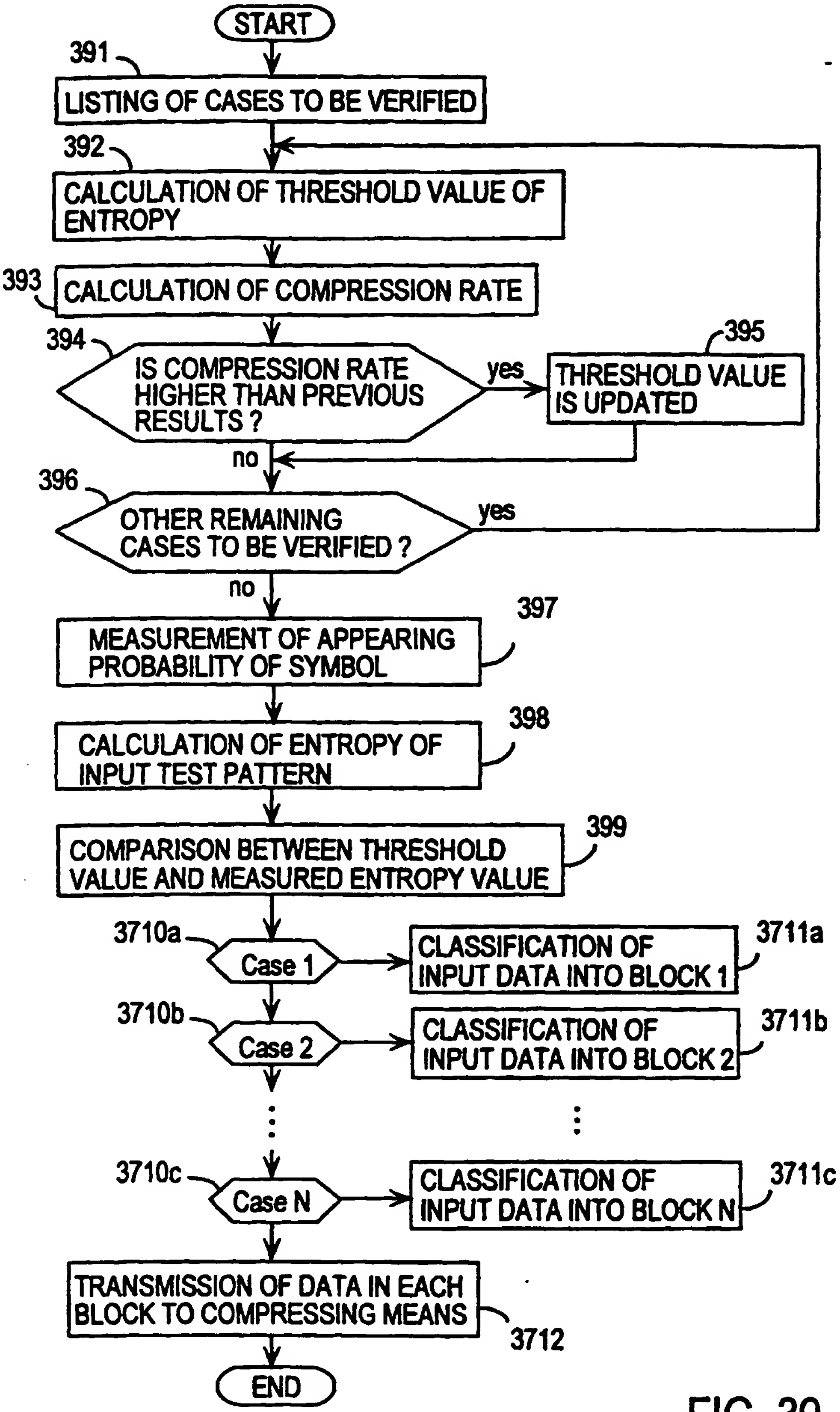

START
391
LISTING OF CASES TO BE VERIFIED
392
CALCULATION OF THRESHOLD VALUE OF ENTROPY
393
CALCULATION OF COMPRESSION RATE
394
IS COMPRESSION RATE HIGHER THAN PREVIOUS RESULTS ?
yes
395
THRESHOLD VALUE IS UPDATED
no
396
OTHER REMAINING CASES TO BE VERIFIED ?
yes
no
397
MEASUREMENT OF APPEARING PROBABILITY OF SYMBOL
398
CALCULATION OF ENTROPY OF INPUT TEST PATTERN
399
COMPARISON BETWEEN THRESHOLD VALUE AND MEASURED ENTROPY VALUE
3710a
Case 1
CLASSIFICATION OF INPUT DATA INTO BLOCK 1
3711a
3710b
Case 2
CLASSIFICATION OF INPUT DATA INTO BLOCK 2
3711b
3710c
Case N
CLASSIFICATION OF INPUT DATA INTO BLOCK N
3711c
TRANSMISSION OF DATA IN EACH BLOCK TO COMPRESSING MEANS
3712
END

FIG. 39

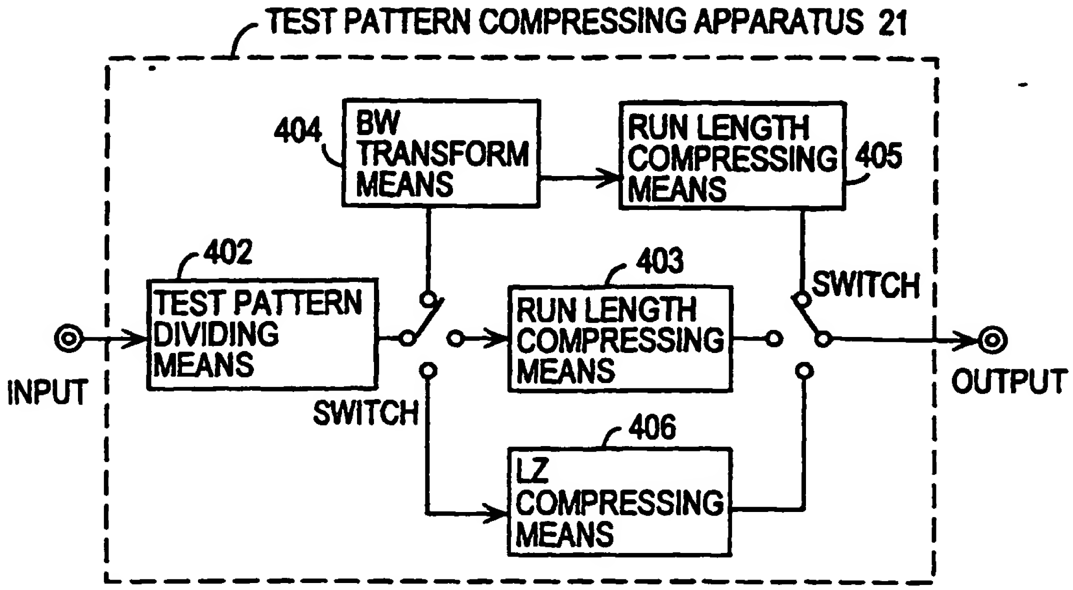

TEST PATTERN COMPRESSING APPARATUS 21
404
BW TRANSFORM MEANS
RUN LENGTH COMPRESSING MEANS
405
402
TEST PATTERN DIVIDING MEANS
403
RUN LENGTH COMPRESSING MEANS
SWITCH
INPUT
SWITCH
OUTPUT
406
LZ COMPRESSING MEANS

FIG. 40

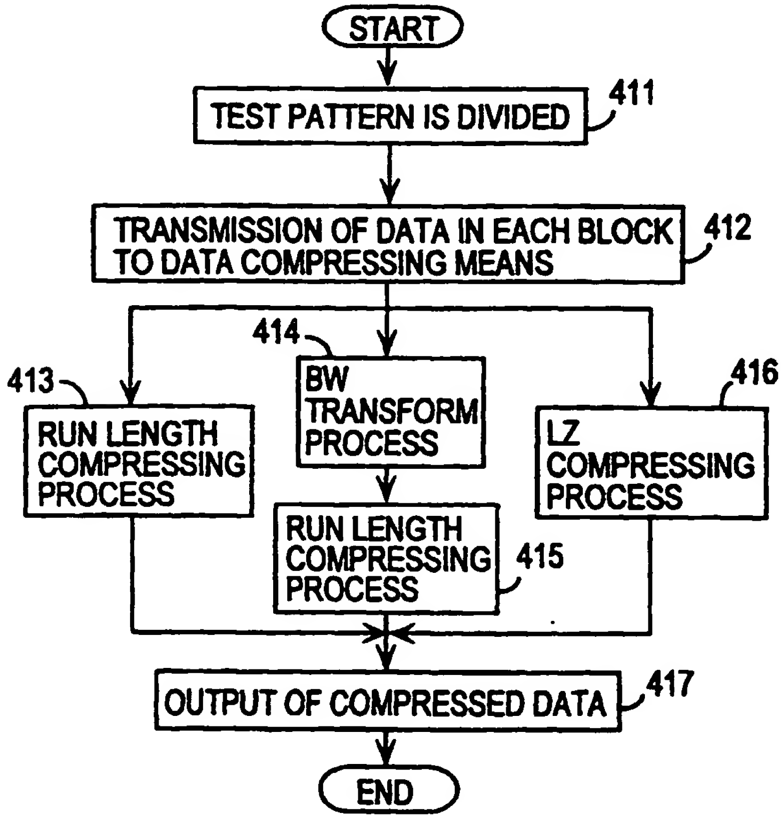

START
TEST PATTERN IS DIVIDED
411
TRANSMISSION OF DATA IN EACH BLOCK TO DATA COMPRESSING MEANS
412
414
BW TRANSFORM PROCESS
413
RUN LENGTH COMPRESSING PROCESS
416
LZ COMPRESSING PROCESS
RUN LENGTH COMPRESSING PROCESS
415
OUTPUT OF COMPRESSED DATA
417
END

FIG. 41

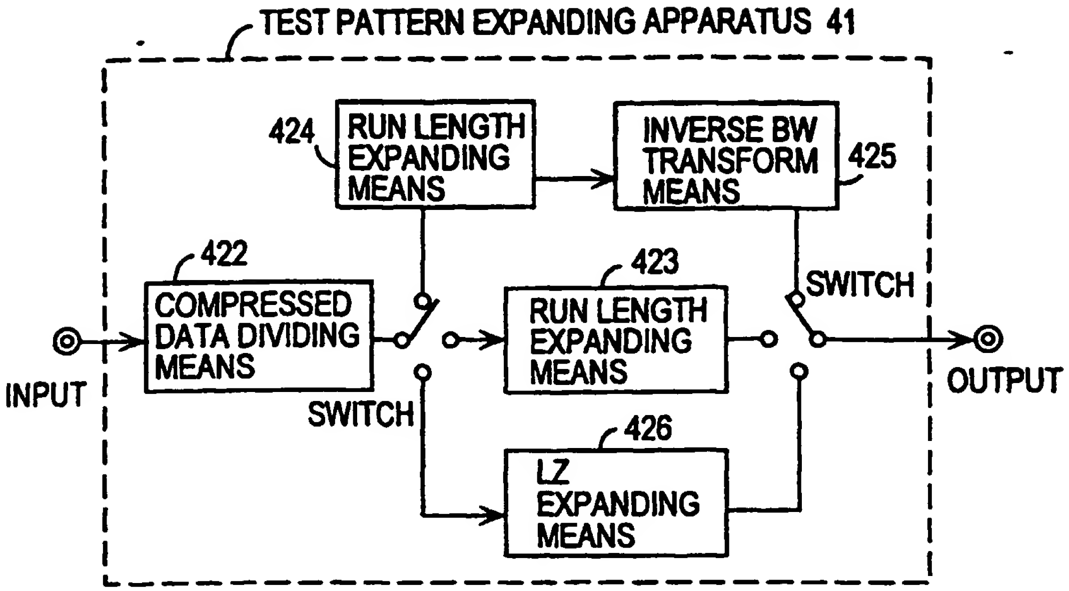

TEST PATTERN EXPANDING APPARATUS 41
424
RUN LENGTH EXPANDING MEANS
INVERSE BW TRANSFORM MEANS
425
422
COMPRESSED DATA DIVIDING MEANS
423
RUN LENGTH EXPANDING MEANS
SWITCH
INPUT
SWITCH
OUTPUT
426
LZ EXPANDING MEANS

FIG. 42

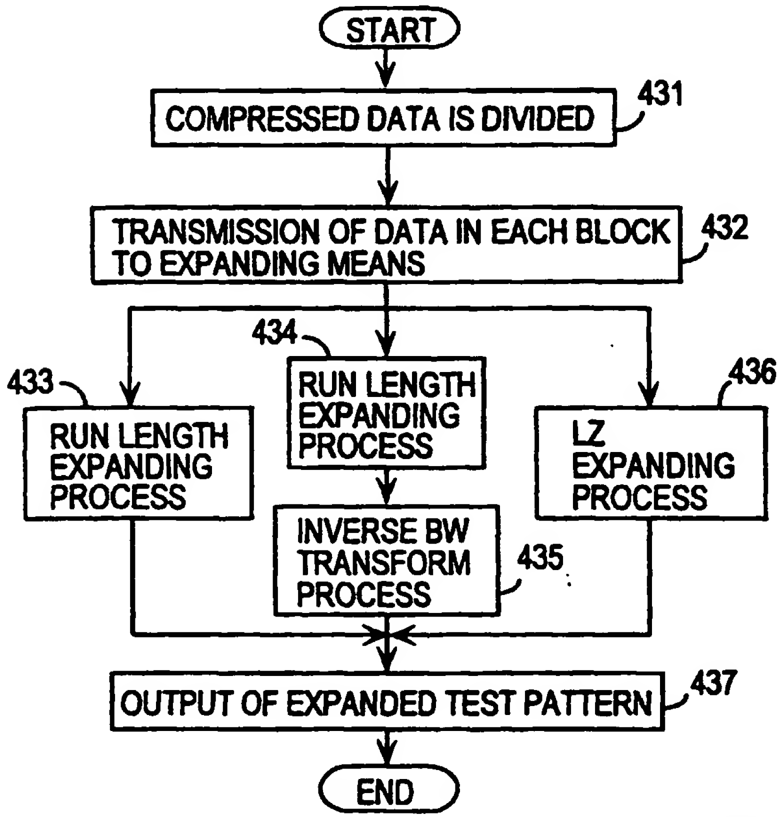

START
COMPRESSED DATA IS DIVIDED
431
TRANSMISSION OF DATA IN EACH BLOCK TO EXPANDING MEANS
432
434
433
436
RUN LENGTH EXPANDING PROCESS
RUN LENGTH EXPANDING PROCESS
LZ EXPANDING PROCESS
INVERSE BW TRANSFORM PROCESS
435
OUTPUT OF EXPANDED TEST PATTERN
437
END

FIG. 43

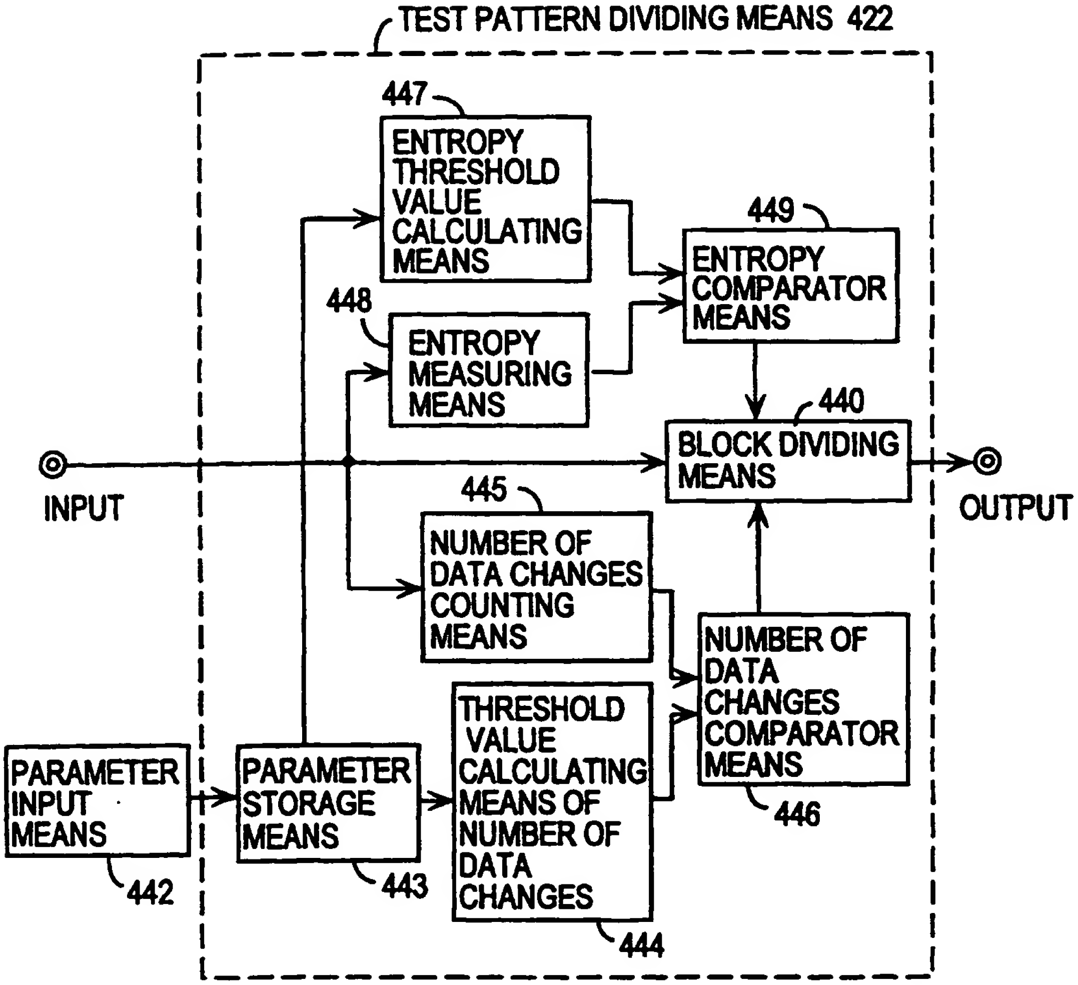

TEST PATTERN DIVIDING MEANS 422
447
ENTROPY THRESHOLD VALUE CALCULATING MEANS
449
ENTROPY COMPARATOR MEANS
448
ENTROPY MEASURING MEANS
440
BLOCK DIVIDING MEANS
INPUT
OUTPUT
445
NUMBER OF DATA CHANGES COUNTING MEANS
NUMBER OF DATA CHANGES COMPARATOR MEANS
446
THRESHOLD VALUE CALCULATING MEANS OF NUMBER OF DATA CHANGES
444
PARAMETER INPUT MEANS
442
PARAMETER STORAGE MEANS
443

FIG. 44

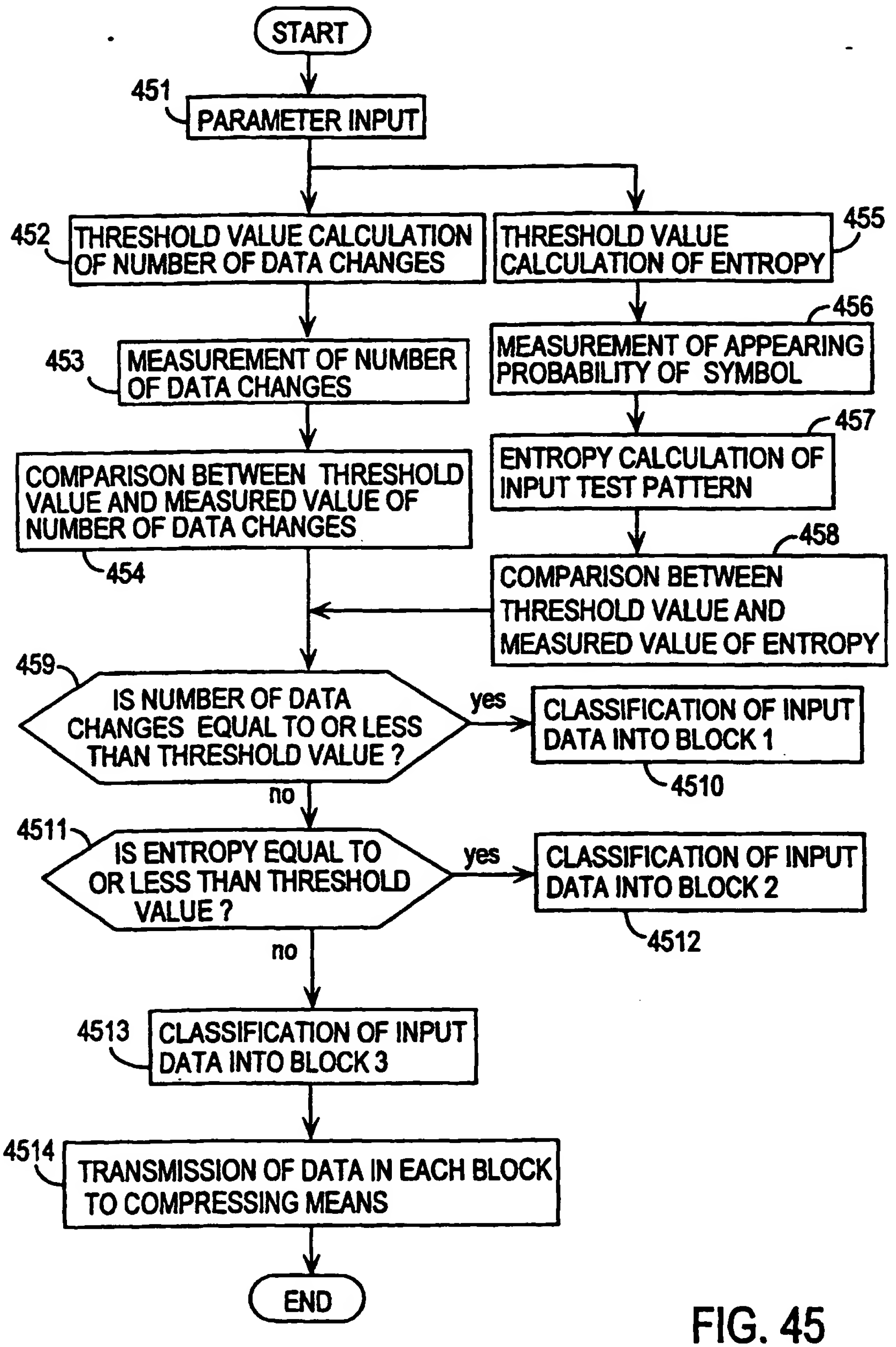

START
451
PARAMETER INPUT
452
THRESHOLD VALUE CALCULATION OF NUMBER OF DATA CHANGES
455
THRESHOLD VALUE CALCULATION OF ENTROPY
453
MEASUREMENT OF NUMBER OF DATA CHANGES
456
MEASUREMENT OF APPEARING PROBABILITY OF SYMBOL
457
ENTROPY CALCULATION OF INPUT TEST PATTERN
COMPARISON BETWEEN THRESHOLD VALUE AND MEASURED VALUE OF NUMBER OF DATA CHANGES
454
458
COMPARISON BETWEEN THRESHOLD VALUE AND MEASURED VALUE OF ENTROPY
459
IS NUMBER OF DATA CHANGES EQUAL TO OR LESS THAN THRESHOLD VALUE ?
yes
CLASSIFICATION OF INPUT DATA INTO BLOCK 1
4510
no
4511
IS ENTROPY EQUAL TO OR LESS THAN THRESHOLD VALUE ?
yes
CLASSIFICATION OF INPUT DATA INTO BLOCK 2
4512
no
4513
CLASSIFICATION OF INPUT DATA INTO BLOCK 3
4514
TRANSMISSION OF DATA IN EACH BLOCK TO COMPRESSING MEANS
END

FIG. 45

FIG. 46

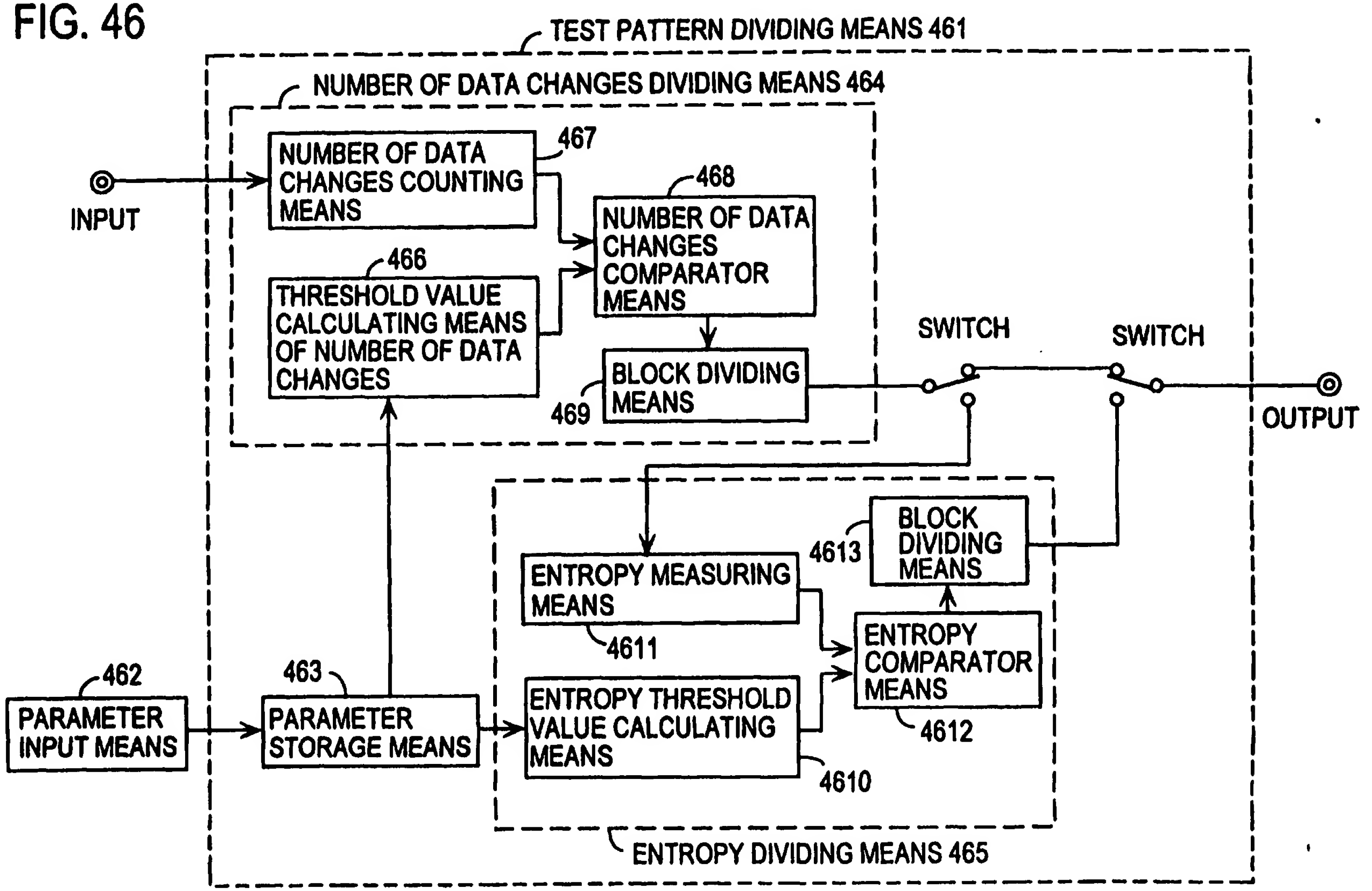
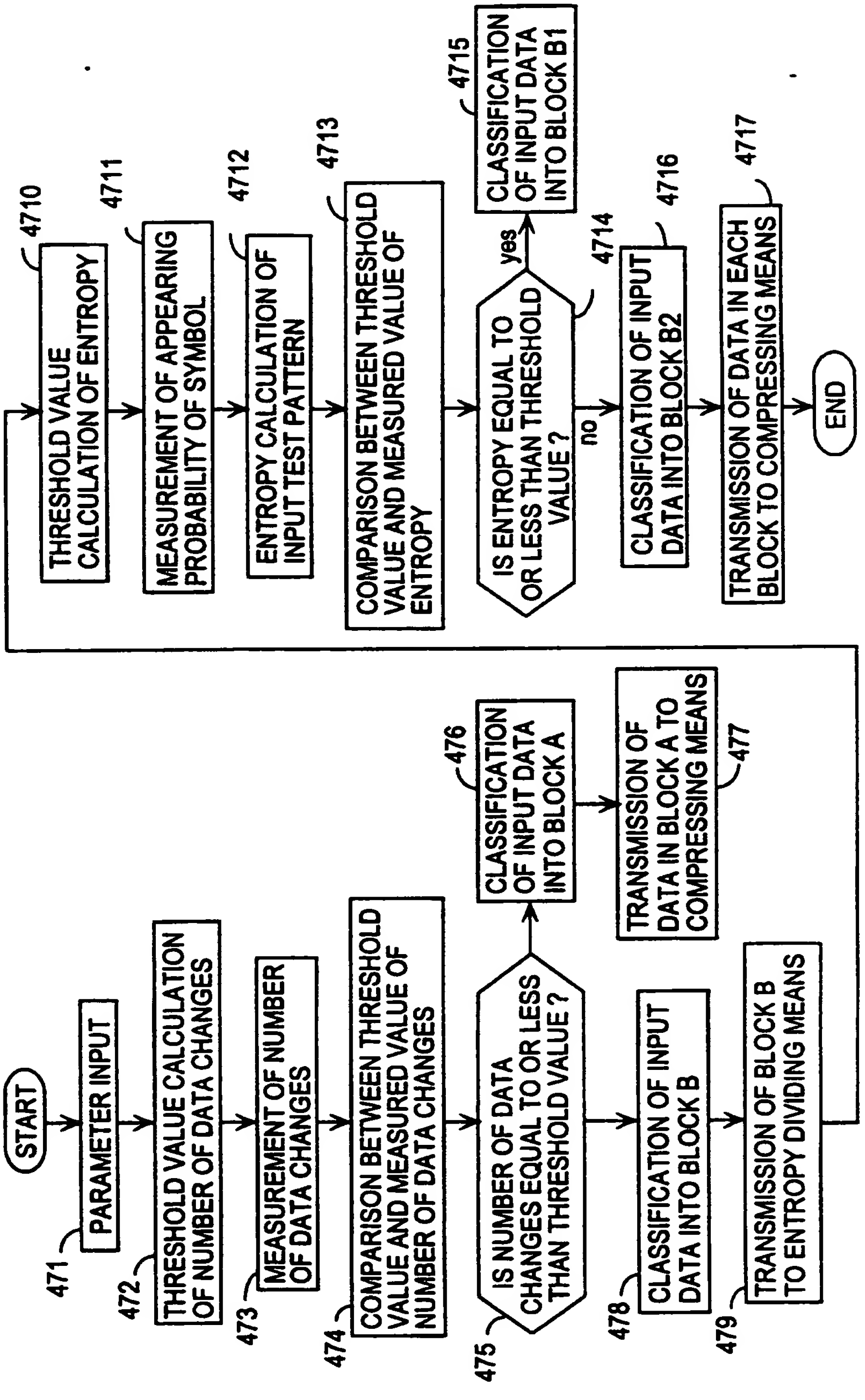

TEST PATTERN DIVIDING MEANS 461
NUMBER OF DATA CHANGES DIVIDING MEANS 464
INPUT
NUMBER OF DATA CHANGES COUNTING MEANS
467
466
THRESHOLD VALUE CALCULATING MEANS OF NUMBER OF DATA CHANGES
468
NUMBER OF DATA CHANGES COMPARATOR MEANS
469
BLOCK DIVIDING MEANS
SWITCH
SWITCH
OUTPUT
462
PARAMETER INPUT MEANS
463
PARAMETER STORAGE MEANS
ENTROPY MEASURING MEANS
4611
ENTROPY THRESHOLD VALUE CALCULATING MEANS
4610
4613
BLOCK DIVIDING MEANS
ENTROPY COMPARATOR MEANS
4612
ENTROPY DIVIDING MEANS 465

FIG. 47


START
471
PARAMETER INPUT
472
THRESHOLD VALUE CALCULATION OF NUMBER OF DATA CHANGES
473
MEASUREMENT OF NUMBER OF DATA CHANGES
474
COMPARISON BETWEEN THRESHOLD VALUE AND MEASURED VALUE OF NUMBER OF DATA CHANGES
475
IS NUMBER OF DATA CHANGES EQUAL TO OR LESS THAN THRESHOLD VALUE ?
476
CLASSIFICATION OF INPUT DATA INTO BLOCK A
477
TRANSMISSION OF DATA IN BLOCK A TO COMPRESSING MEANS
478
CLASSIFICATION OF INPUT DATA INTO BLOCK B
479
TRANSMISSION OF BLOCK B TO ENTROPY DIVIDING MEANS
4710
THRESHOLD VALUE CALCULATION OF ENTROPY
4711
MEASUREMENT OF APPEARING PROBABILITY OF SYMBOL
4712
ENTROPY CALCULATION OF INPUT TEST PATTERN
4713
COMPARISON BETWEEN THRESHOLD VALUE AND MEASURED VALUE OF ENTROPY
4714
IS ENTROPY EQUAL TO OR LESS THAN THRESHOLD VALUE ?
yes
no
4715
CLASSIFICATION OF INPUT DATA INTO BLOCK B1
4716
CLASSIFICATION OF INPUT DATA INTO BLOCK B2
4717
TRANSMISSION OF DATA IN EACH BLOCK TO COMPRESSING MEANS
END

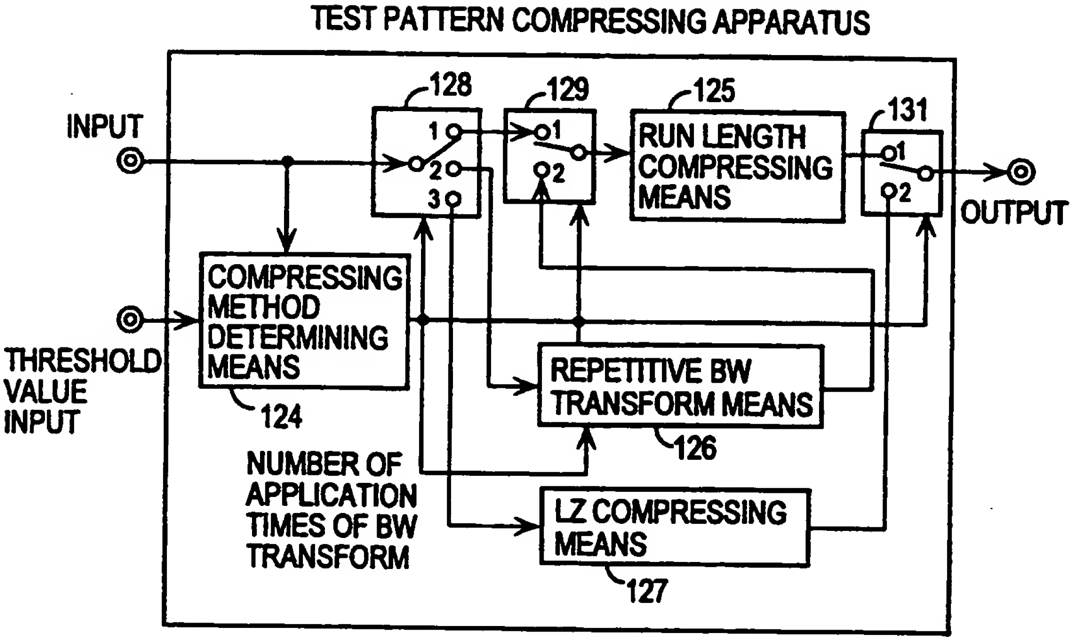

TEST PATTERN COMPRESSING APPARATUS
INPUT
THRESHOLD VALUE INPUT
RUN LENGTH COMPRESSING MEANS
OUTPUT
COMPRESSING METHOD DETERMINING MEANS
REPETITIVE BW TRANSFORM MEANS
NUMBER OF APPLICATION TIMES OF BW TRANSFORM
LZ COMPRESSING MEANS

FIG. 48

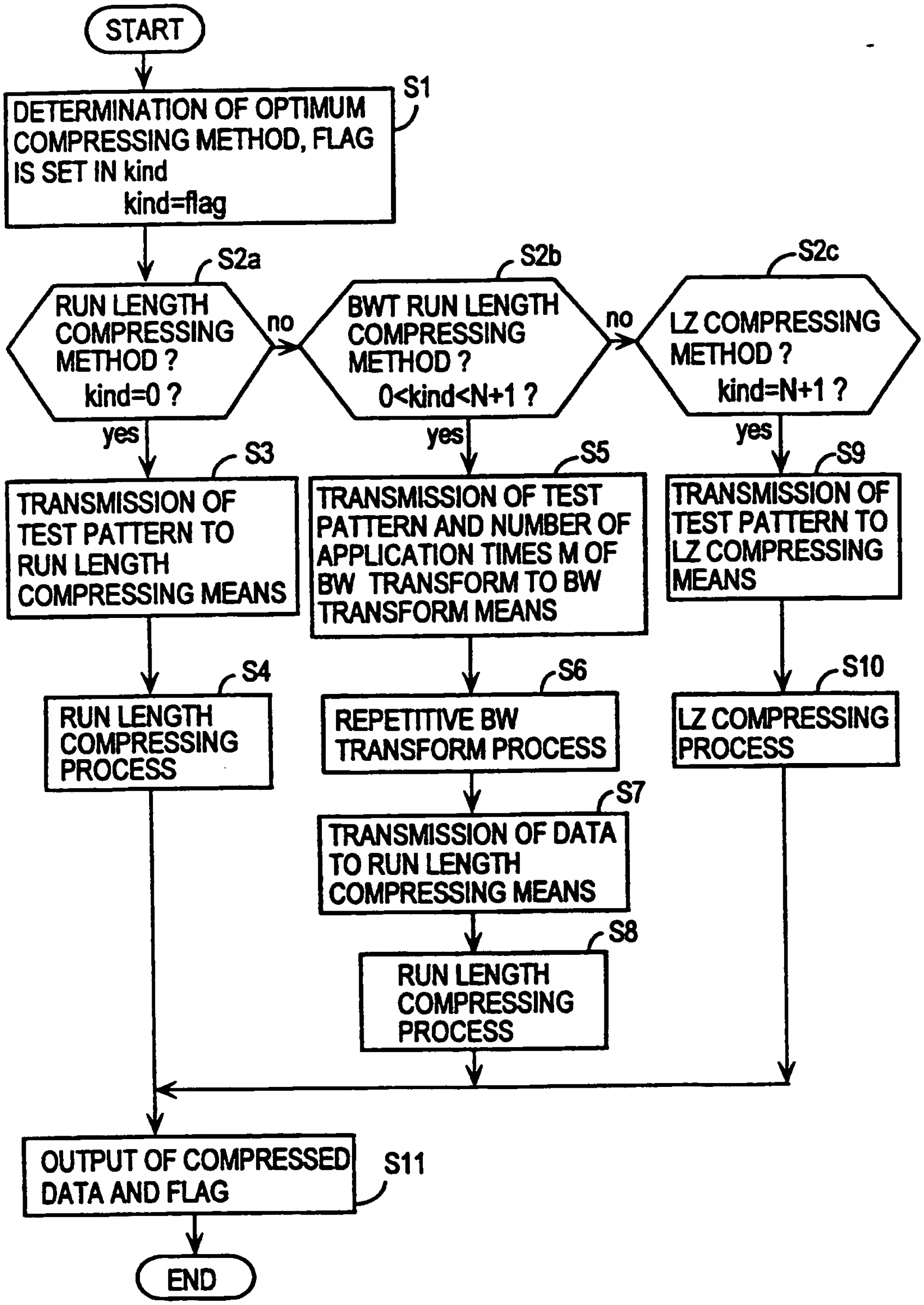

START
DETERMINATION OF OPTIMUM COMPRESSING METHOD, FLAG IS SET IN kind
kind=flag
S1
RUN LENGTH COMPRESSING METHOD ?
kind=0 ?
S2a
no
BWT RUN LENGTH COMPRESSING METHOD ?
0<kind<N+1 ?
S2b
no
LZ COMPRESSING METHOD ?
kind=N+1 ?
S2c
yes
yes
yes
TRANSMISSION OF TEST PATTERN TO RUN LENGTH COMPRESSING MEANS
S3
TRANSMISSION OF TEST PATTERN AND NUMBER OF APPLICATION TIMES M OF BW TRANSFORM TO BW TRANSFORM MEANS
S5
TRANSMISSION OF TEST PATTERN TO LZ COMPRESSING MEANS
S9
RUN LENGTH COMPRESSING PROCESS
S4
REPETITIVE BW TRANSFORM PROCESS
S6
LZ COMPRESSING PROCESS
S10
TRANSMISSION OF DATA TO RUN LENGTH COMPRESSING MEANS
S7
RUN LENGTH COMPRESSING PROCESS
S8
OUTPUT OF COMPRESSED DATA AND FLAG
S11
END

FIG.49

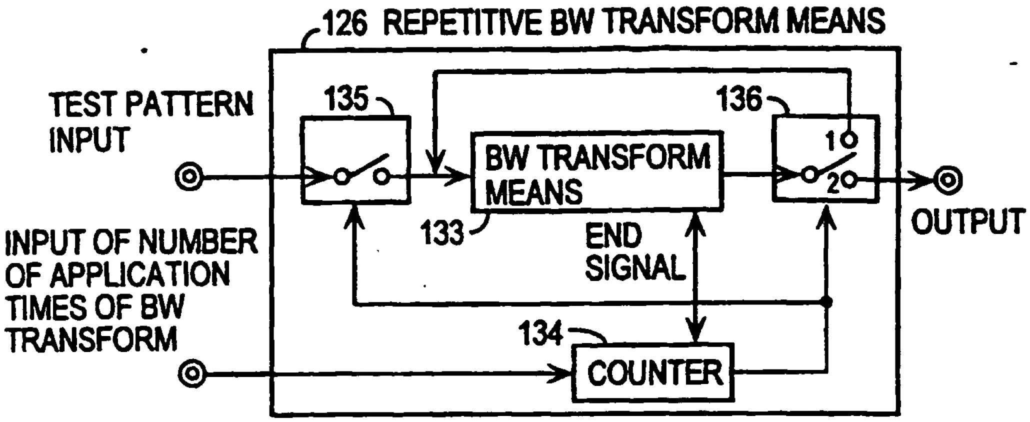

126 REPETITIVE BW TRANSFORM MEANS
TEST PATTERN INPUT
135
136
1
2
BW TRANSFORM MEANS
133
END SIGNAL
OUTPUT
INPUT OF NUMBER OF APPLICATION TIMES OF BW TRANSFORM
134
COUNTER

FIG. 50

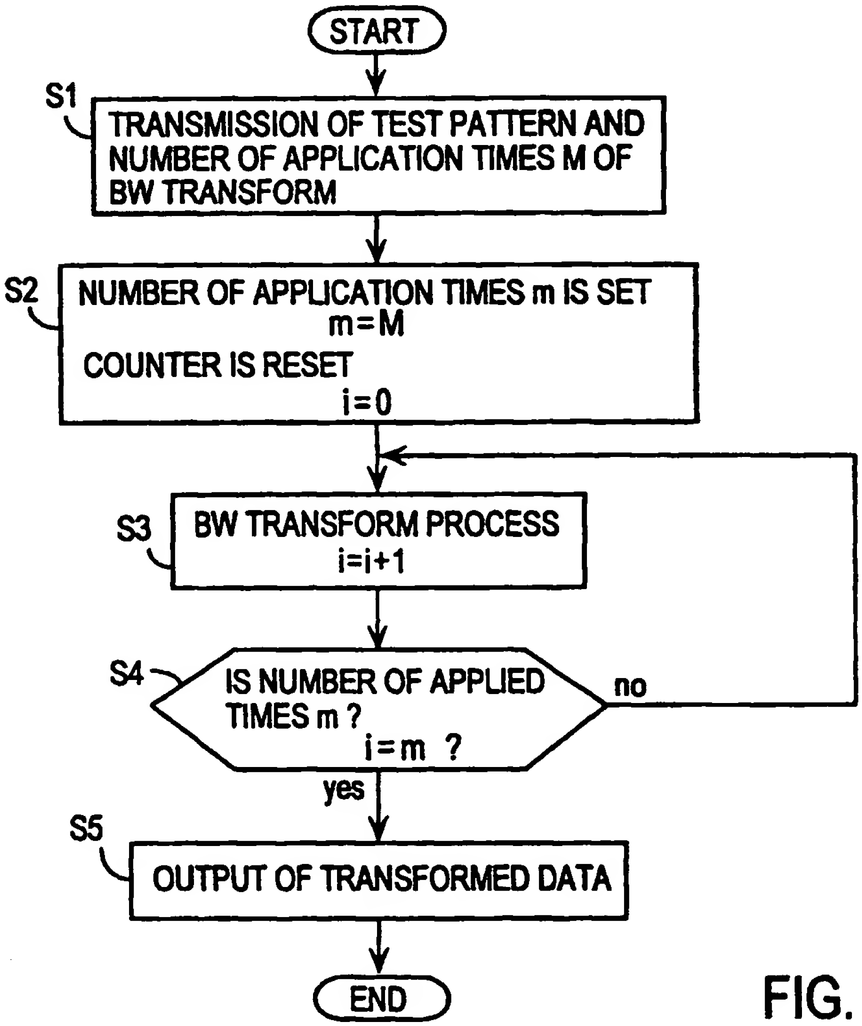

START
S1
TRANSMISSION OF TEST PATTERN AND NUMBER OF APPLICATION TIMES M OF BW TRANSFORM
S2
NUMBER OF APPLICATION TIMES m IS SET
m=M
COUNTER IS RESET
i=0
S3
BW TRANSFORM PROCESS
i=i+1
S4
IS NUMBER OF APPLIED TIMES m ?
i=m ?
no
yes
S5
OUTPUT OF TRANSFORMED DATA
END

FIG. 51

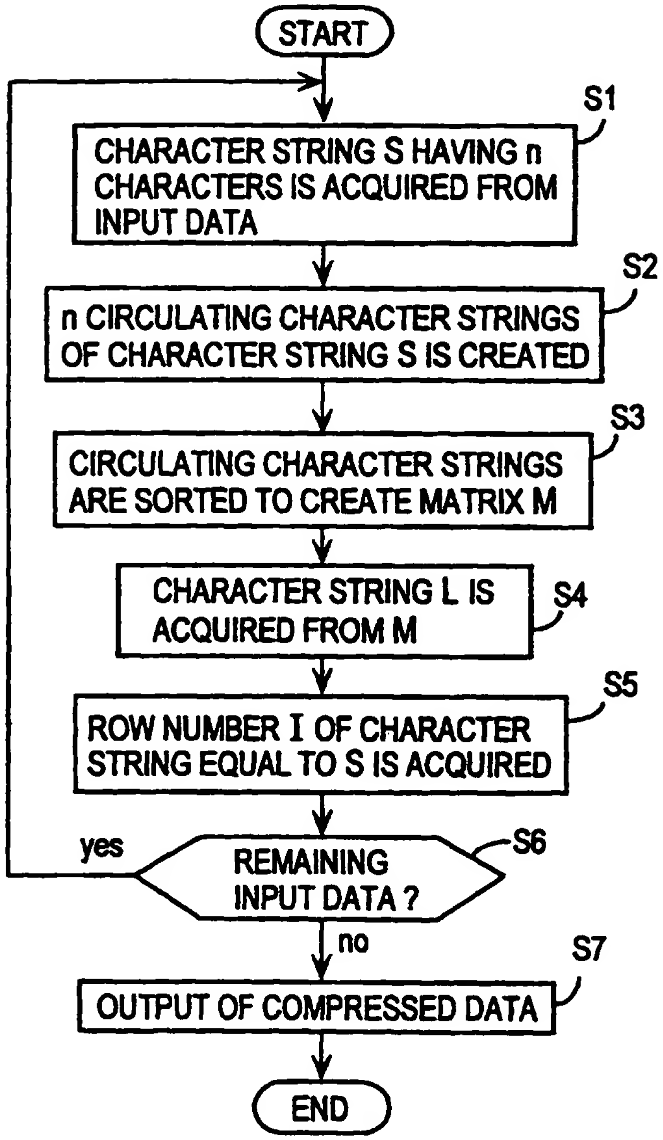

START
S1
CHARACTER STRING S HAVING n CHARACTERS IS ACQUIRED FROM INPUT DATA
S2
n CIRCULATING CHARACTER STRINGS OF CHARACTER STRING S IS CREATED
S3
CIRCULATING CHARACTER STRINGS ARE SORTED TO CREATE MATRIX M
CHARACTER STRING L IS ACQUIRED FROM M
S4
S5
ROW NUMBER I OF CHARACTER STRING EQUAL TO S IS ACQUIRED
yes
REMAINING INPUT DATA ?
S6
no
S7
OUTPUT OF COMPRESSED DATA
END

FIG. 52

FIG. 53

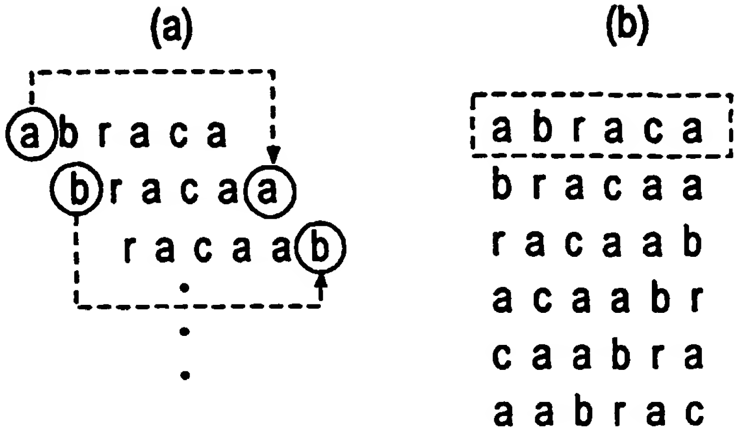

(a)
a b r a c a
b r a c a a
r a c a a b
(b)
a b r a c a
b r a c a a
r a c a a b
a c a a b r
c a a b r a
a a b r a c

FIG. 54

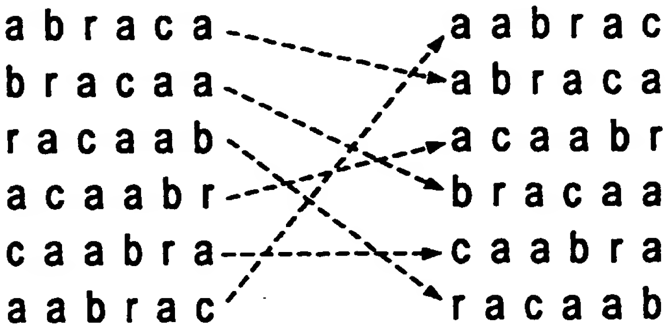
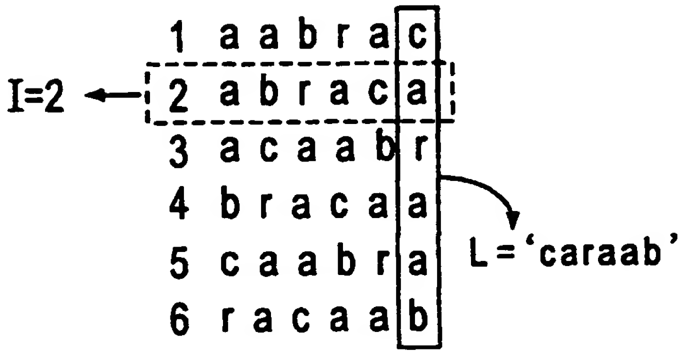

a b r a c a
b r a c a a
r a c a a b
a c a a b r
c a a b r a
a a b r a c
a a b r a c
a b r a c a
a c a a b r
b r a c a a
c a a b r a
r a c a a b

FIG. 55


1 a a b r a c
I=2
2 a b r a c a
3 a c a a b r
4 b r a c a a
5 c a a b r a
6 r a c a a b
L = 'caraab'

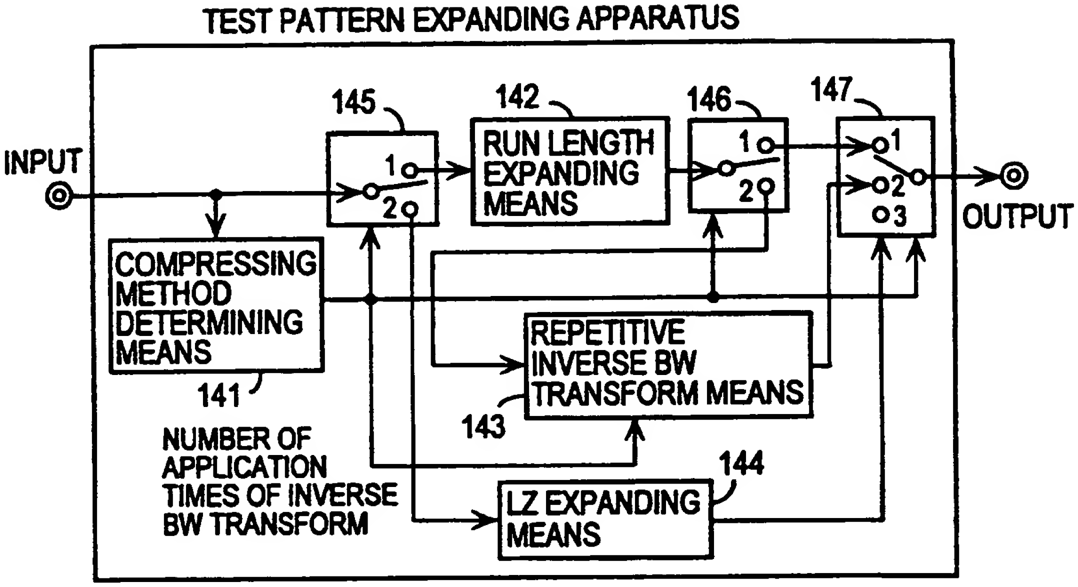

TEST PATTERN EXPANDING APPARATUS
INPUT
OUTPUT
RUN LENGTH EXPANDING MEANS
COMPRESSING METHOD DETERMINING MEANS
REPETITIVE INVERSE BW TRANSFORM MEANS
NUMBER OF APPLICATION TIMES OF INVERSE BW TRANSFORM
LZ EXPANDING MEANS

FIG. 56

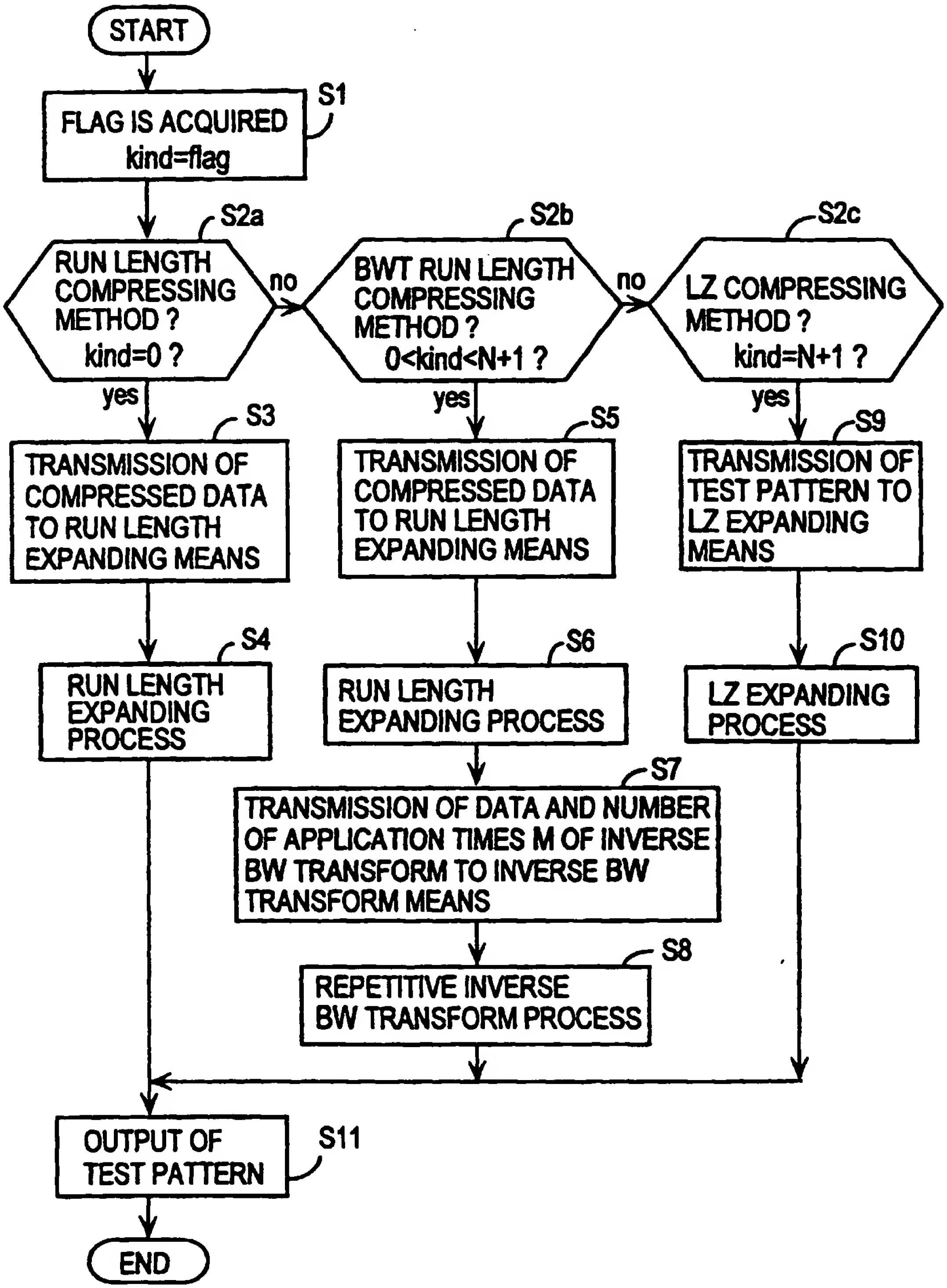

START
FLAG IS ACQUIRED kind=flag
S1
S2a
RUN LENGTH COMPRESSING METHOD ? kind=0 ?
no
S2b
BWT RUN LENGTH COMPRESSING METHOD ? 0<kind<N+1 ?
no
S2c
LZ COMPRESSING METHOD ? kind=N+1 ?
yes
yes
yes
S3
TRANSMISSION OF COMPRESSED DATA TO RUN LENGTH EXPANDING MEANS
S5
TRANSMISSION OF COMPRESSED DATA TO RUN LENGTH EXPANDING MEANS
S9
TRANSMISSION OF TEST PATTERN TO LZ EXPANDING MEANS
S4
RUN LENGTH EXPANDING PROCESS
S6
RUN LENGTH EXPANDING PROCESS
S10
LZ EXPANDING PROCESS
S7
TRANSMISSION OF DATA AND NUMBER OF APPLICATION TIMES M OF INVERSE BW TRANSFORM TO INVERSE BW TRANSFORM MEANS
S8
REPETITIVE INVERSE BW TRANSFORM PROCESS
OUTPUT OF TEST PATTERN
S11
END

FIG. 57

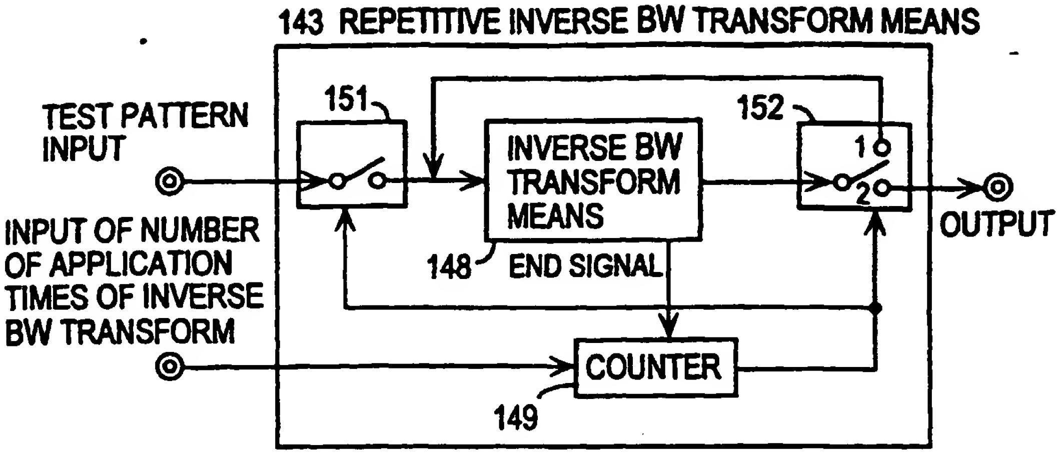

143 REPETITIVE INVERSE BW TRANSFORM MEANS
TEST PATTERN INPUT
INPUT OF NUMBER OF APPLICATION TIMES OF INVERSE BW TRANSFORM
151
INVERSE BW TRANSFORM MEANS
148
END SIGNAL
152
1
2
OUTPUT
COUNTER
149

FIG. 58

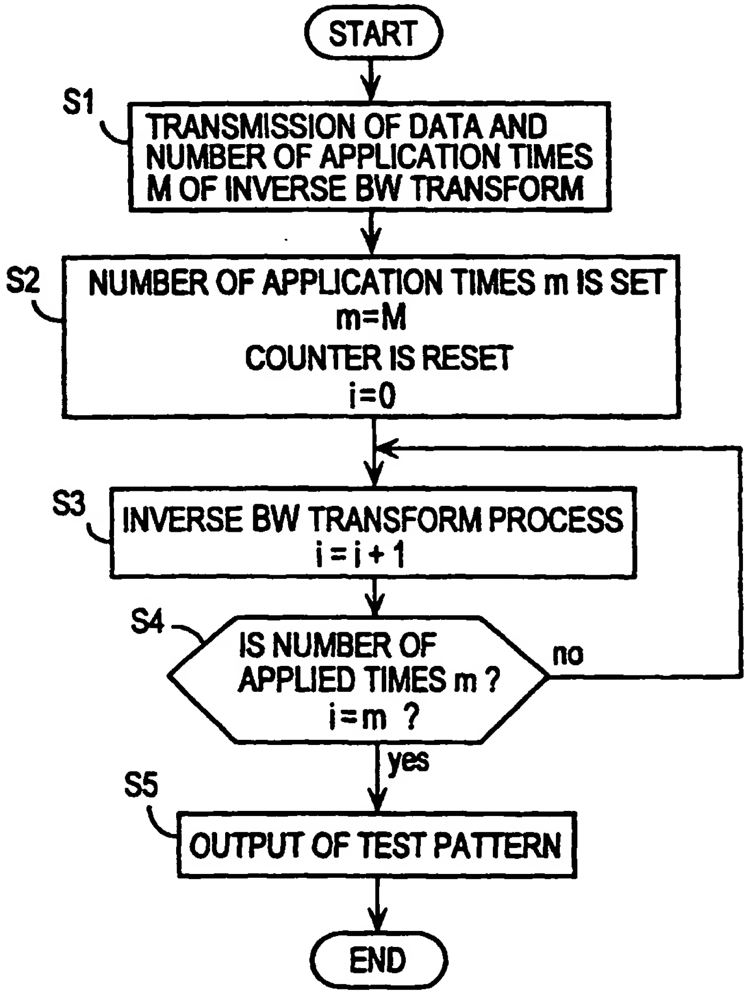

START
S1
TRANSMISSION OF DATA AND NUMBER OF APPLICATION TIMES M OF INVERSE BW TRANSFORM
S2
NUMBER OF APPLICATION TIMES m IS SET
m=M
COUNTER IS RESET
i=0
S3
INVERSE BW TRANSFORM PROCESS
i=i+1
S4
IS NUMBER OF APPLIED TIMES m ?
i=m ?
no
yes
S5
OUTPUT OF TEST PATTERN
END

FIG. 59

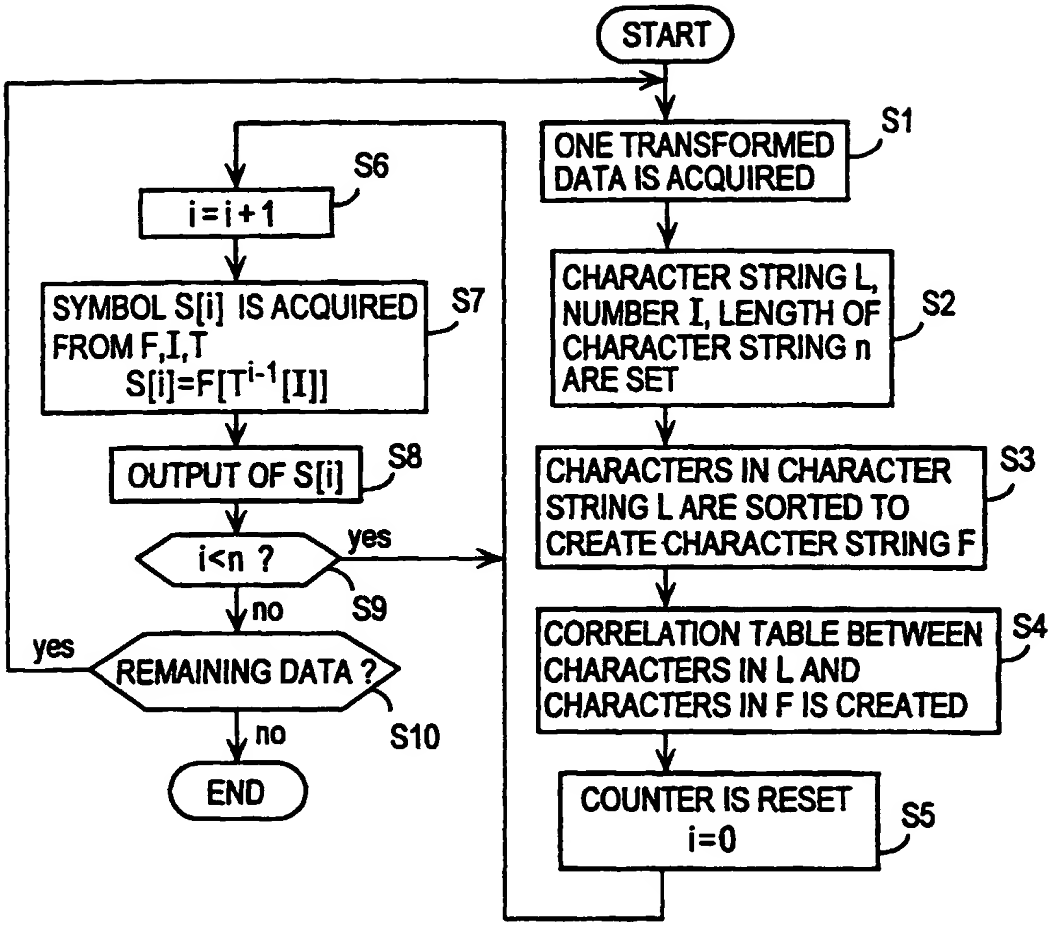

START
ONE TRANSFORMED DATA IS ACQUIRED
S1
CHARACTER STRING L, NUMBER I, LENGTH OF CHARACTER STRING n ARE SET
S2
CHARACTERS IN CHARACTER STRING L ARE SORTED TO CREATE CHARACTER STRING F
S3
CORRELATION TABLE BETWEEN CHARACTERS IN L AND CHARACTERS IN F IS CREATED
S4
COUNTER IS RESET i=0
S5
i=i+1
S6
SYMBOL S[i] IS ACQUIRED FROM F,I,T S[i]=F[T^{i-1}[I]]
S7
OUTPUT OF S[i]
S8
i<n ?
yes
no
S9
REMAINING DATA ?
yes
no
S10
END

FIG. 60

FIG. 61

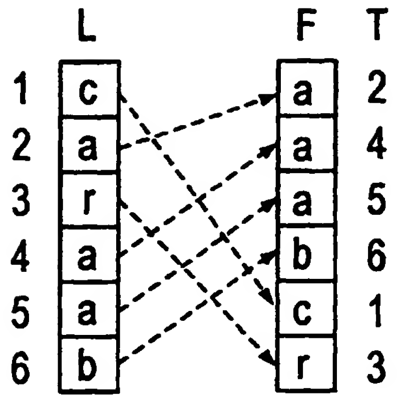

L
F
T
1
2
3
4
5
6
c
a
r
a
a
b
a
a
a
b
c
r
2
4
5
6
1
3

FIG. 62

i	$T^{i-1}[I]$	$S[i]=F[T^{i-1}[I]]$
1	2	a
2	4	b
3	6	r
4	3	a
5	5	c
6	1	a

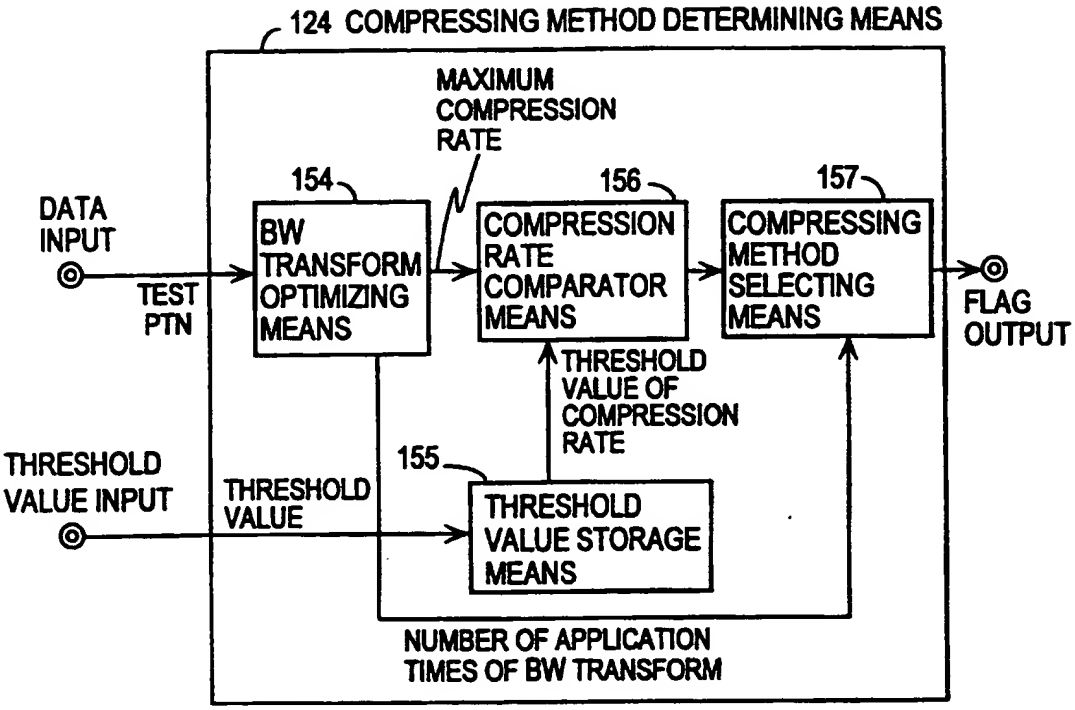

124 COMPRESSING METHOD DETERMINING MEANS
MAXIMUM COMPRESSION RATE
154
156
157
DATA INPUT
TEST PTN
BW TRANSFORM OPTIMIZING MEANS
COMPRESSION RATE COMPARATOR MEANS
COMPRESSING METHOD SELECTING MEANS
FLAG OUTPUT
THRESHOLD VALUE OF COMPRESSION RATE
155
THRESHOLD VALUE INPUT
THRESHOLD VALUE
THRESHOLD VALUE STORAGE MEANS
NUMBER OF APPLICATION TIMES OF BW TRANSFORM

FIG. 63

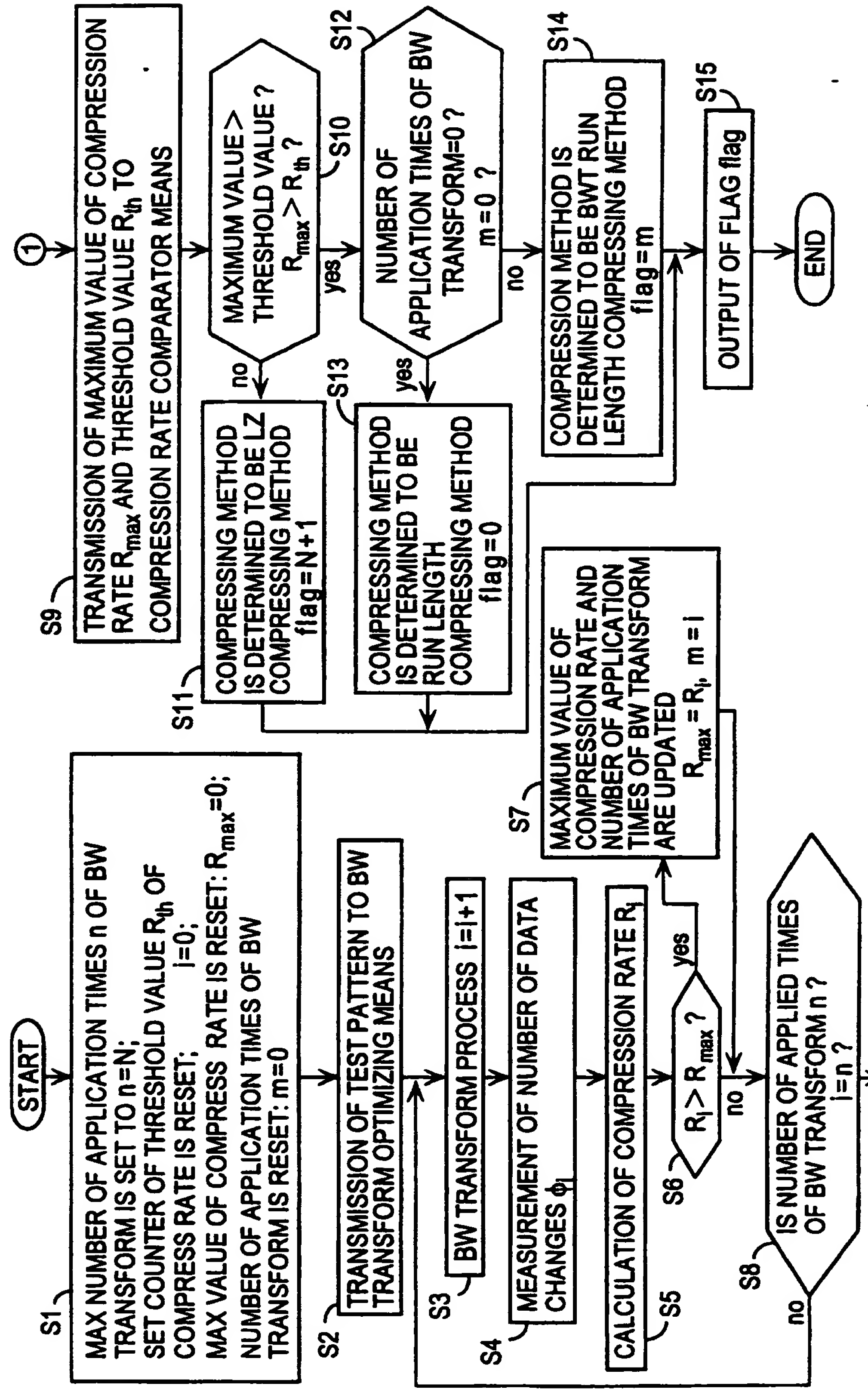

START
S1
MAX NUMBER OF APPLICATION TIMES n OF BW TRANSFORM IS SET TO n=N;
SET COUNTER OF THRESHOLD VALUE R_{th} OF COMPRESS RATE IS RESET; I=0;
MAX VALUE OF COMPRESS RATE IS RESET: R_{max}=0;
NUMBER OF APPLICATION TIMES OF BW TRANSFORM IS RESET: m=0
S2
TRANSMISSION OF TEST PATTERN TO BW TRANSFORM OPTIMIZING MEANS
S3
BW TRANSFORM PROCESS I=I+1
S4
MEASUREMENT OF NUMBER OF DATA CHANGES ϕ_I
S5
CALCULATION OF COMPRESSION RATE R_I
S6
$R_I > R_{max}$?
yes
no
S7
MAXIMUM VALUE OF COMPRESSION RATE AND NUMBER OF APPLICATION TIMES OF BW TRANSFORM ARE UPDATED
$R_{max} = R_I$, m = I
S8
IS NUMBER OF APPLIED TIMES OF BW TRANSFORM n ?
I=n ?
yes
no
①
①
S9
TRANSMISSION OF MAXIMUM VALUE OF COMPRESSION RATE R_{max} AND THRESHOLD VALUE R_{th} TO COMPRESSION RATE COMPARATOR MEANS
S10
MAXIMUM VALUE > THRESHOLD VALUE ?
$R_{max} > R_{th}$?
yes
no
S11
COMPRESSING METHOD IS DETERMINED TO BE LZ COMPRESSING METHOD
flag=N+1
S12
NUMBER OF APPLICATION TIMES OF BW TRANSFORM=0 ?
m=0 ?
yes
no
S13
COMPRESSING METHOD IS DETERMINED TO BE RUN LENGTH COMPRESSING METHOD
flag=0
S14
COMPRESSION METHOD IS DETERMINED TO BE BWT RUN LENGTH COMPRESSING METHOD
flag=m
S15
OUTPUT OF FLAG flag
END

FIG. 64

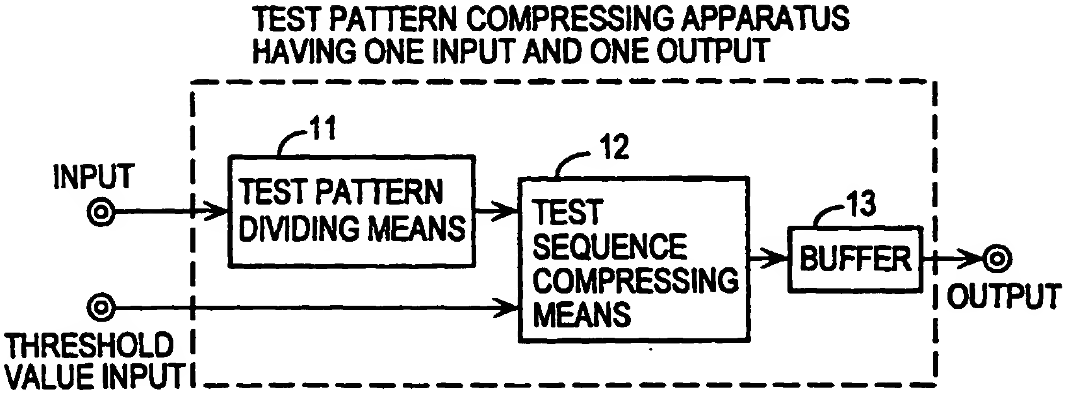

TEST PATTERN COMPRESSING APPARATUS
HAVING ONE INPUT AND ONE OUTPUT
INPUT
THRESHOLD
VALUE INPUT
11
TEST PATTERN
DIVIDING MEANS
12
TEST
SEQUENCE
COMPRESSING
MEANS
13
BUFFER
OUTPUT

FIG. 65

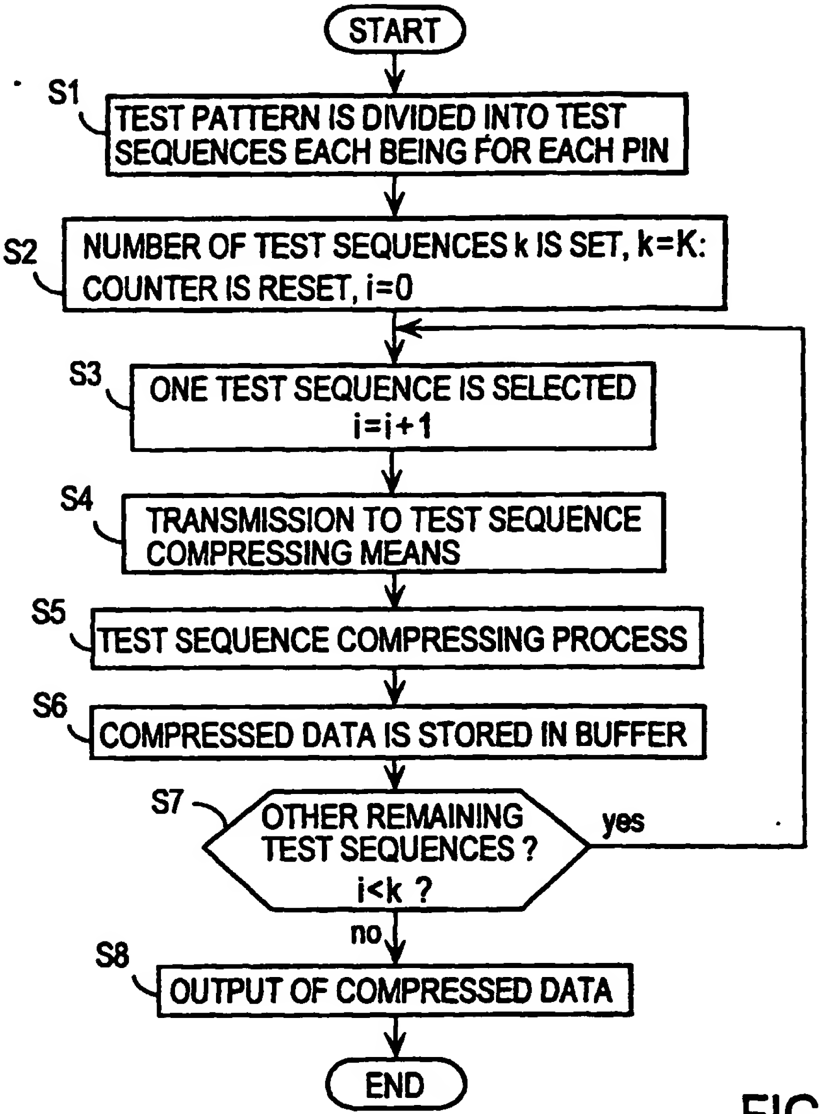

START
S1
TEST PATTERN IS DIVIDED INTO TEST SEQUENCES EACH BEING FOR EACH PIN
S2
NUMBER OF TEST SEQUENCES k IS SET, k=K: COUNTER IS RESET, i=0
S3
ONE TEST SEQUENCE IS SELECTED i=i+1
S4
TRANSMISSION TO TEST SEQUENCE COMPRESSING MEANS
S5
TEST SEQUENCE COMPRESSING PROCESS
S6
COMPRESSED DATA IS STORED IN BUFFER
S7
OTHER REMAINING TEST SEQUENCES ? i<k ?
yes
no
S8
OUTPUT OF COMPRESSED DATA
END

FIG. 66

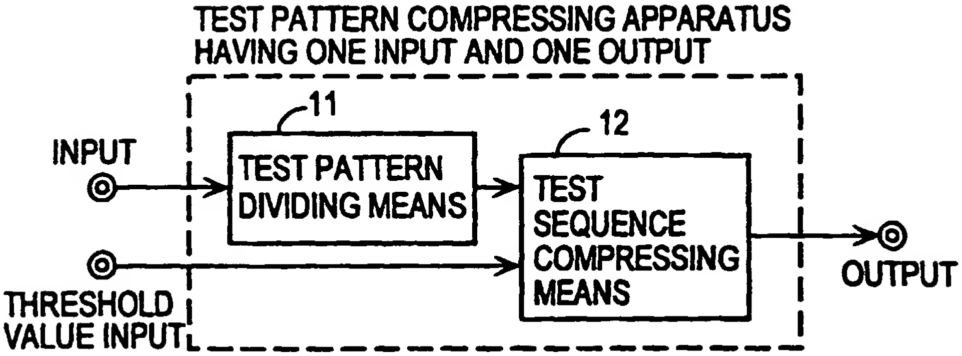

TEST PATTERN COMPRESSING APPARATUS HAVING ONE INPUT AND ONE OUTPUT
11
12
INPUT
TEST PATTERN DIVIDING MEANS
TEST SEQUENCE COMPRESSING MEANS
OUTPUT
THRESHOLD VALUE INPUT

FIG. 67

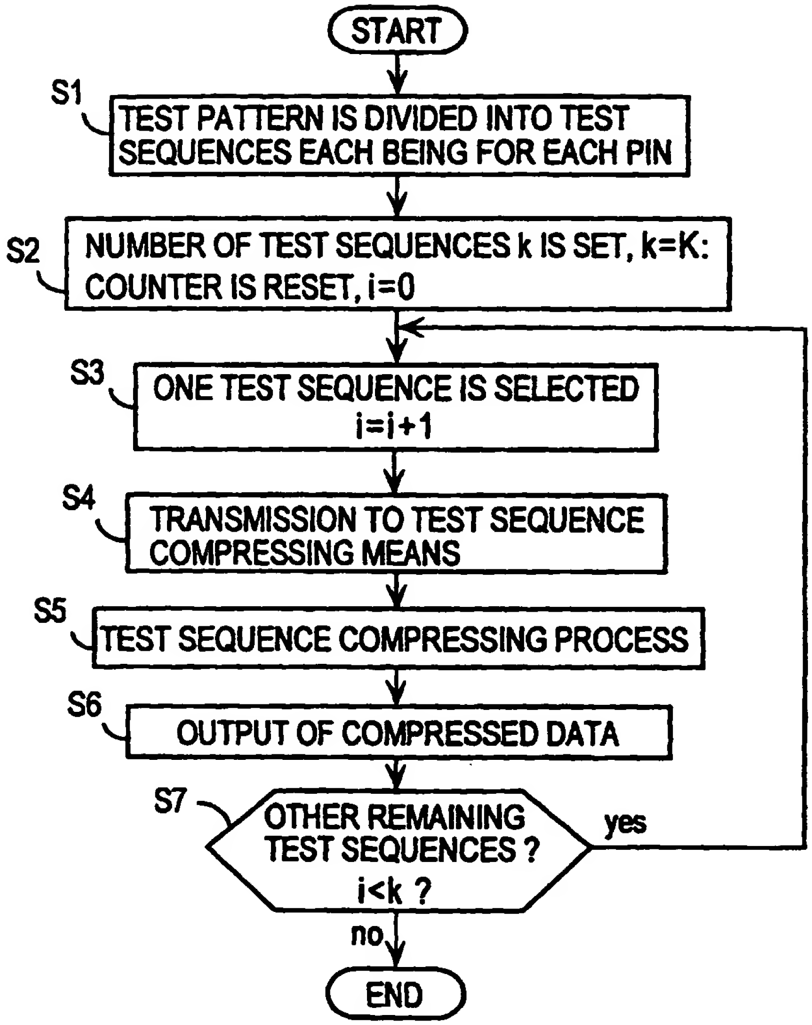

START
S1
TEST PATTERN IS DIVIDED INTO TEST SEQUENCES EACH BEING FOR EACH PIN
S2
NUMBER OF TEST SEQUENCES k IS SET, k=K: COUNTER IS RESET, i=0
S3
ONE TEST SEQUENCE IS SELECTED i=i+1
S4
TRANSMISSION TO TEST SEQUENCE COMPRESSING MEANS
S5
TEST SEQUENCE COMPRESSING PROCESS
S6
OUTPUT OF COMPRESSED DATA
S7
OTHER REMAINING TEST SEQUENCES ? i<k ?
yes
no
END

FIG. 68

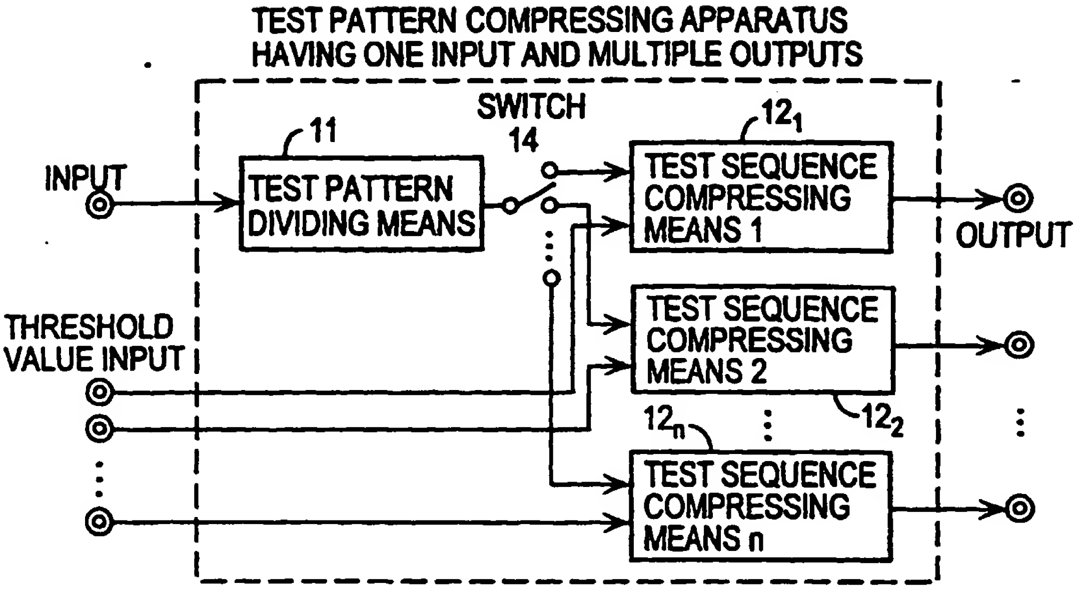

TEST PATTERN COMPRESSING APPARATUS
HAVING ONE INPUT AND MULTIPLE OUTPUTS
INPUT
11
TEST PATTERN DIVIDING MEANS
SWITCH
14
12_1
TEST SEQUENCE COMPRESSING MEANS 1
OUTPUT
THRESHOLD VALUE INPUT
TEST SEQUENCE COMPRESSING MEANS 2
12_2
12_n
TEST SEQUENCE COMPRESSING MEANS n

FIG. 69

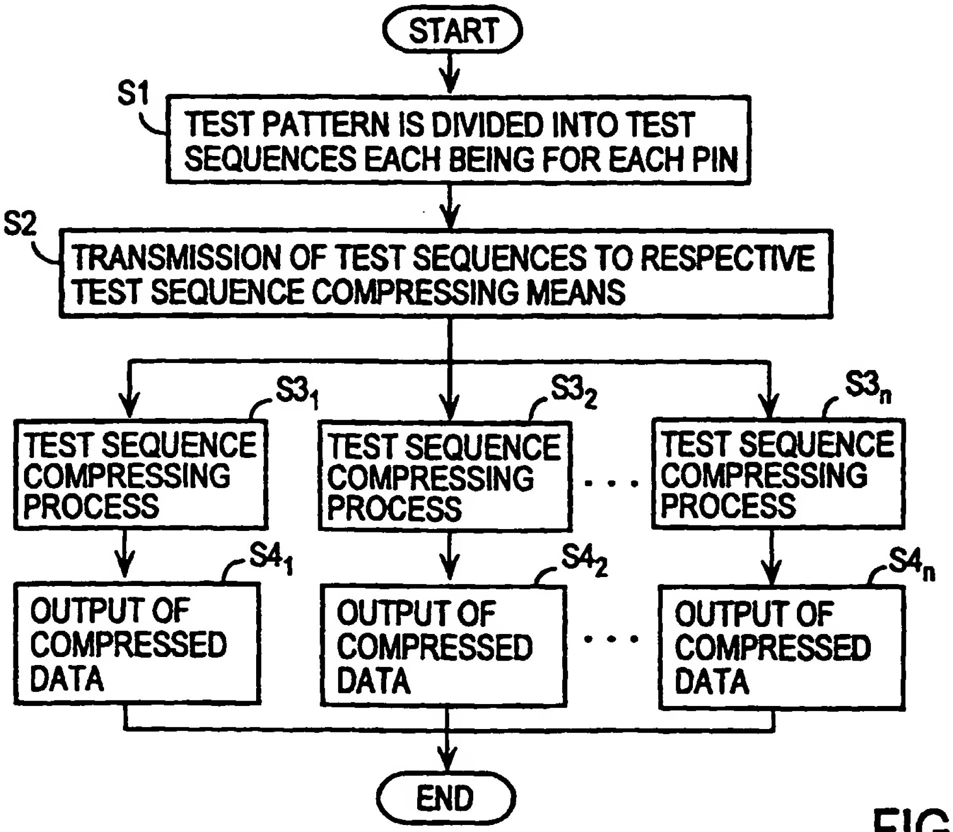

START
S1
TEST PATTERN IS DIVIDED INTO TEST SEQUENCES EACH BEING FOR EACH PIN
S2
TRANSMISSION OF TEST SEQUENCES TO RESPECTIVE TEST SEQUENCE COMPRESSING MEANS
$S3_1$
TEST SEQUENCE COMPRESSING PROCESS
$S3_2$
TEST SEQUENCE COMPRESSING PROCESS
$S3_n$
TEST SEQUENCE COMPRESSING PROCESS
$S4_1$
OUTPUT OF COMPRESSED DATA
$S4_2$
OUTPUT OF COMPRESSED DATA
$S4_n$
OUTPUT OF COMPRESSED DATA
END

FIG. 70

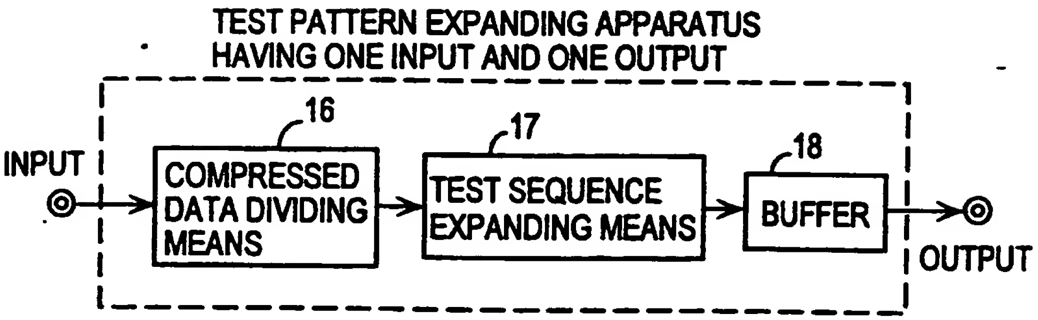

TEST PATTERN EXPANDING APPARATUS
HAVING ONE INPUT AND ONE OUTPUT
INPUT
16
COMPRESSED DATA DIVIDING MEANS
17
TEST SEQUENCE EXPANDING MEANS
18
BUFFER
OUTPUT

FIG. 71

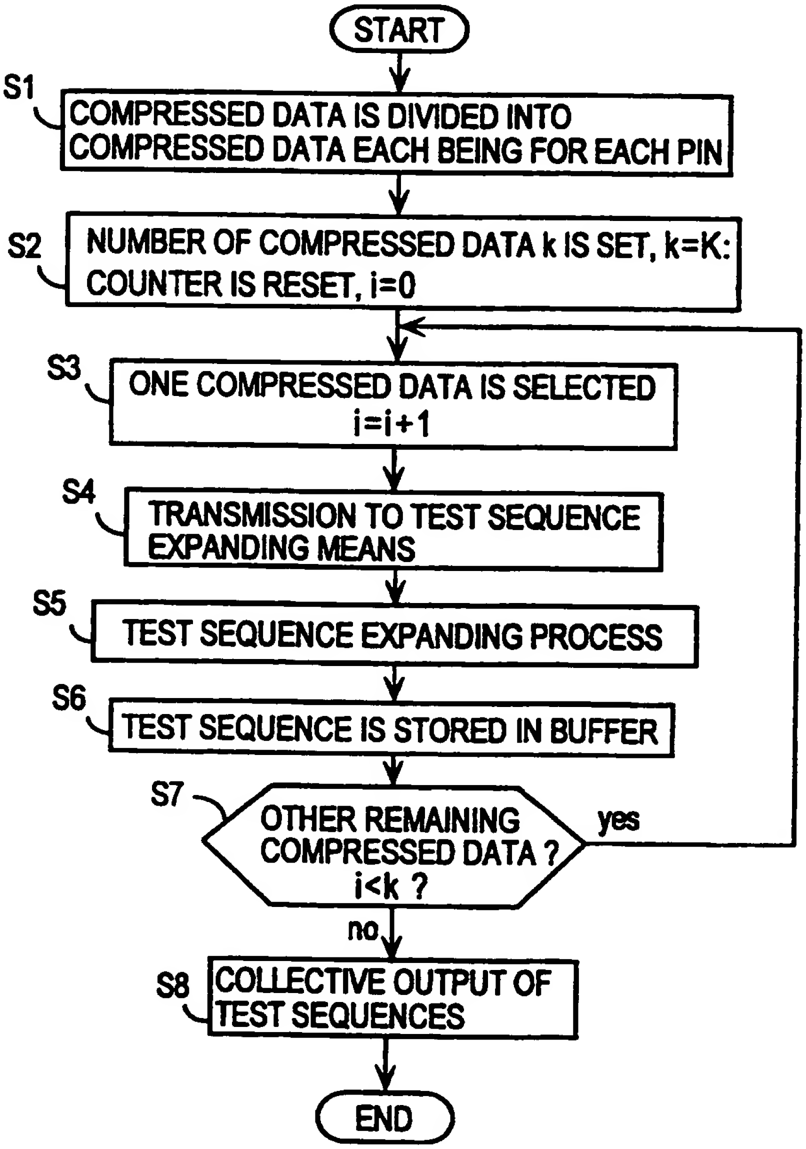

START
S1
COMPRESSED DATA IS DIVIDED INTO COMPRESSED DATA EACH BEING FOR EACH PIN
S2
NUMBER OF COMPRESSED DATA k IS SET, k=K:
COUNTER IS RESET, i=0
S3
ONE COMPRESSED DATA IS SELECTED
i=i+1
S4
TRANSMISSION TO TEST SEQUENCE EXPANDING MEANS
S5
TEST SEQUENCE EXPANDING PROCESS
S6
TEST SEQUENCE IS STORED IN BUFFER
S7
OTHER REMAINING COMPRESSED DATA ?
i<k ?
yes
no
S8
COLLECTIVE OUTPUT OF TEST SEQUENCES
END

FIG. 72

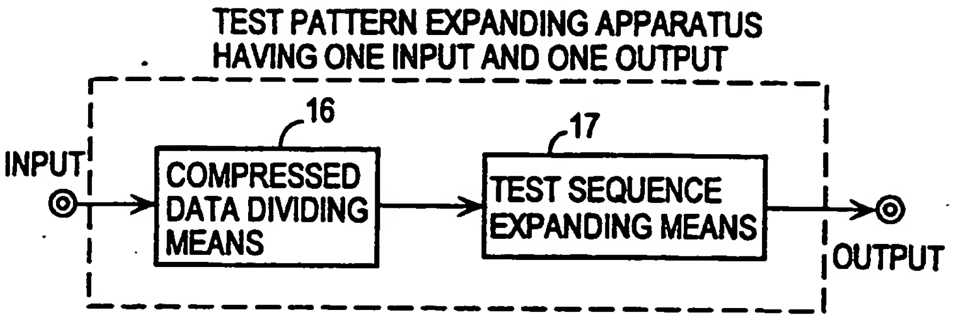

TEST PATTERN EXPANDING APPARATUS
HAVING ONE INPUT AND ONE OUTPUT
INPUT
16
COMPRESSED DATA DIVIDING MEANS
17
TEST SEQUENCE EXPANDING MEANS
OUTPUT

FIG. 73

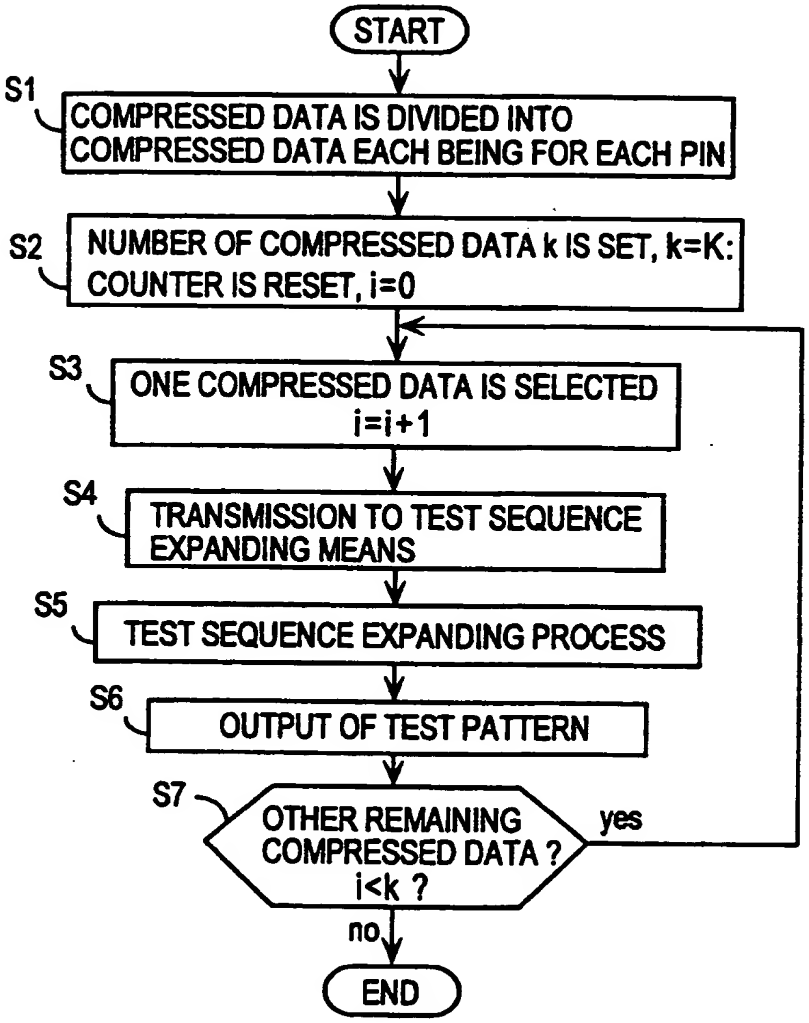

START
S1
COMPRESSED DATA IS DIVIDED INTO COMPRESSED DATA EACH BEING FOR EACH PIN
S2
NUMBER OF COMPRESSED DATA k IS SET, k=K:
COUNTER IS RESET, i=0
S3
ONE COMPRESSED DATA IS SELECTED
i=i+1
S4
TRANSMISSION TO TEST SEQUENCE EXPANDING MEANS
S5
TEST SEQUENCE EXPANDING PROCESS
S6
OUTPUT OF TEST PATTERN
S7
OTHER REMAINING COMPRESSED DATA ?
i<k ?
yes
no
END

FIG. 74

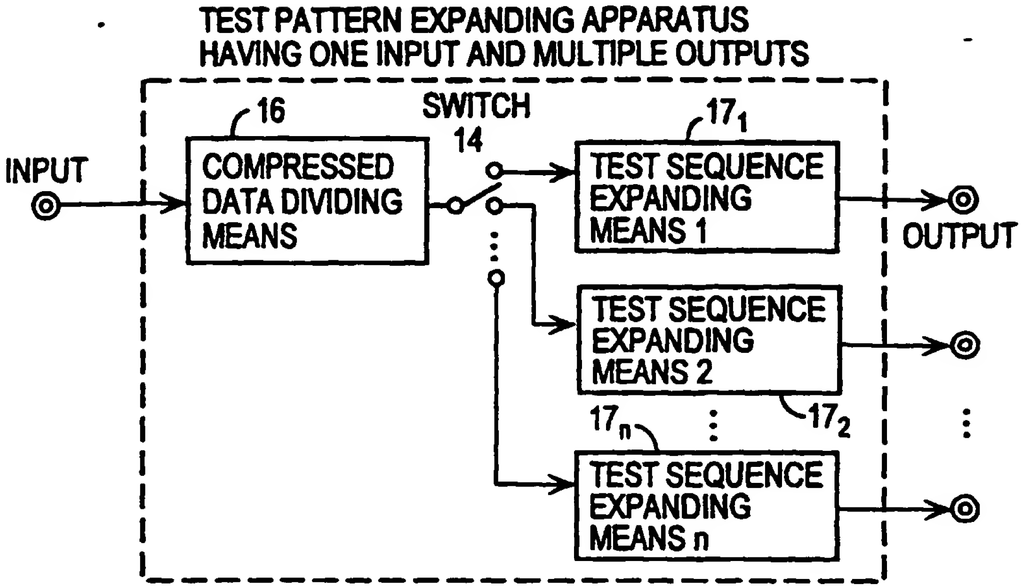

TEST PATTERN EXPANDING APPARATUS HAVING ONE INPUT AND MULTIPLE OUTPUTS
INPUT
16
COMPRESSED DATA DIVIDING MEANS
SWITCH 14
17₁
TEST SEQUENCE EXPANDING MEANS 1
OUTPUT
TEST SEQUENCE EXPANDING MEANS 2
17₂
17ₙ
TEST SEQUENCE EXPANDING MEANS n

FIG. 75

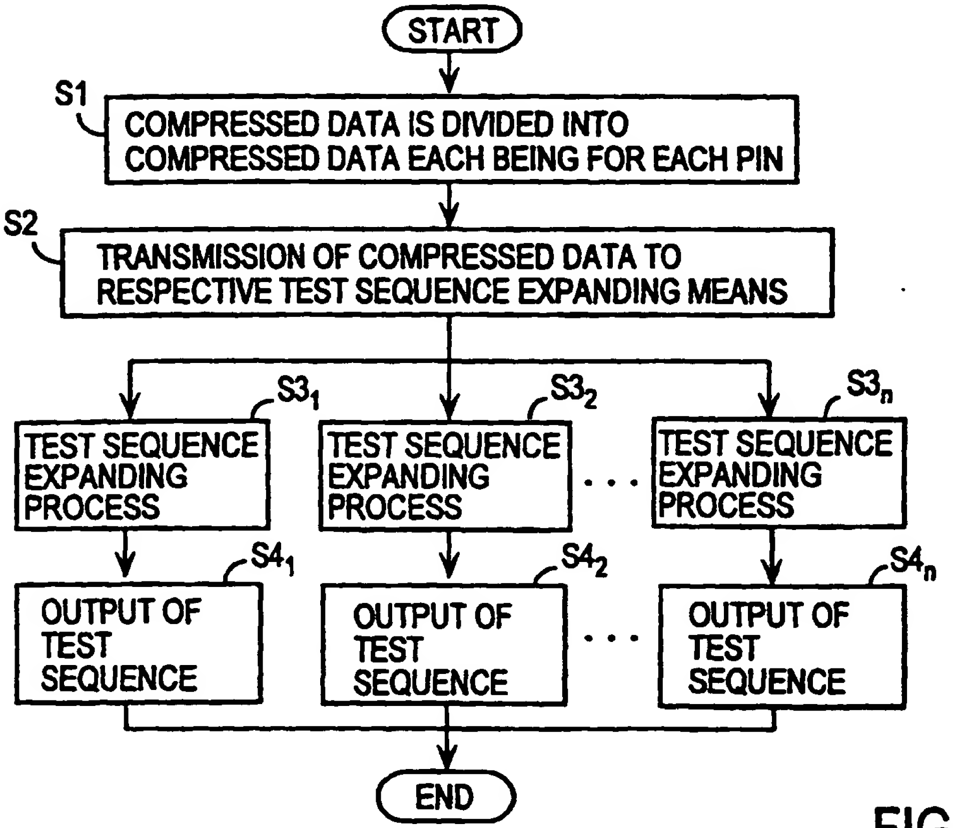

START
S1
COMPRESSED DATA IS DIVIDED INTO COMPRESSED DATA EACH BEING FOR EACH PIN
S2
TRANSMISSION OF COMPRESSED DATA TO RESPECTIVE TEST SEQUENCE EXPANDING MEANS
S3₁
TEST SEQUENCE EXPANDING PROCESS
S3₂
TEST SEQUENCE EXPANDING PROCESS
S3ₙ
TEST SEQUENCE EXPANDING PROCESS
S4₁
OUTPUT OF TEST SEQUENCE
S4₂
OUTPUT OF TEST SEQUENCE
S4ₙ
OUTPUT OF TEST SEQUENCE
END

FIG. 76

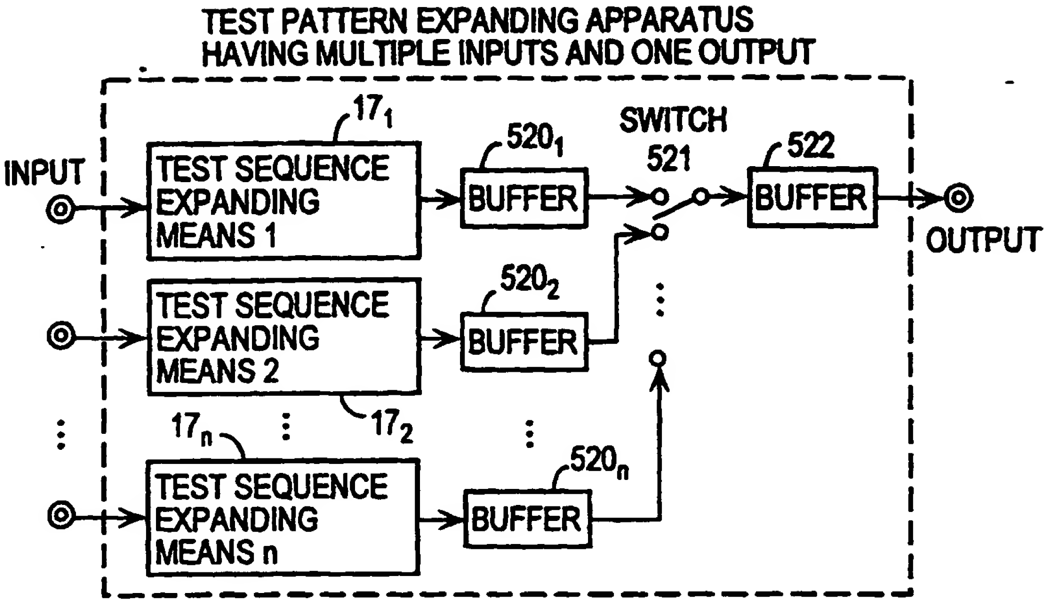

TEST PATTERN EXPANDING APPARATUS HAVING MULTIPLE INPUTS AND ONE OUTPUT
INPUT
17_1
TEST SEQUENCE EXPANDING MEANS 1
520_1
BUFFER
SWITCH 521
522
BUFFER
OUTPUT
TEST SEQUENCE EXPANDING MEANS 2
520_2
BUFFER
17_2
17_n
TEST SEQUENCE EXPANDING MEANS n
520_n
BUFFER

FIG. 77

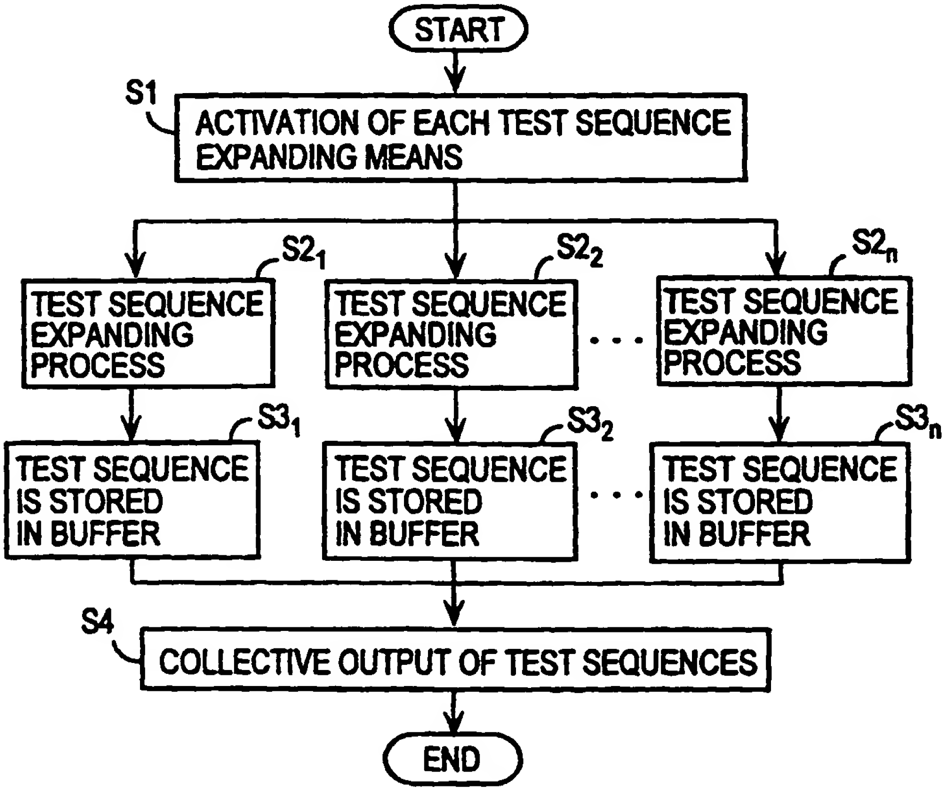

START
S1
ACTIVATION OF EACH TEST SEQUENCE EXPANDING MEANS
$S2_1$
TEST SEQUENCE EXPANDING PROCESS
$S2_2$
TEST SEQUENCE EXPANDING PROCESS
$S2_n$
TEST SEQUENCE EXPANDING PROCESS
$S3_1$
TEST SEQUENCE IS STORED IN BUFFER
$S3_2$
TEST SEQUENCE IS STORED IN BUFFER
$S3_n$
TEST SEQUENCE IS STORED IN BUFFER
S4
COLLECTIVE OUTPUT OF TEST SEQUENCES
END

FIG. 78

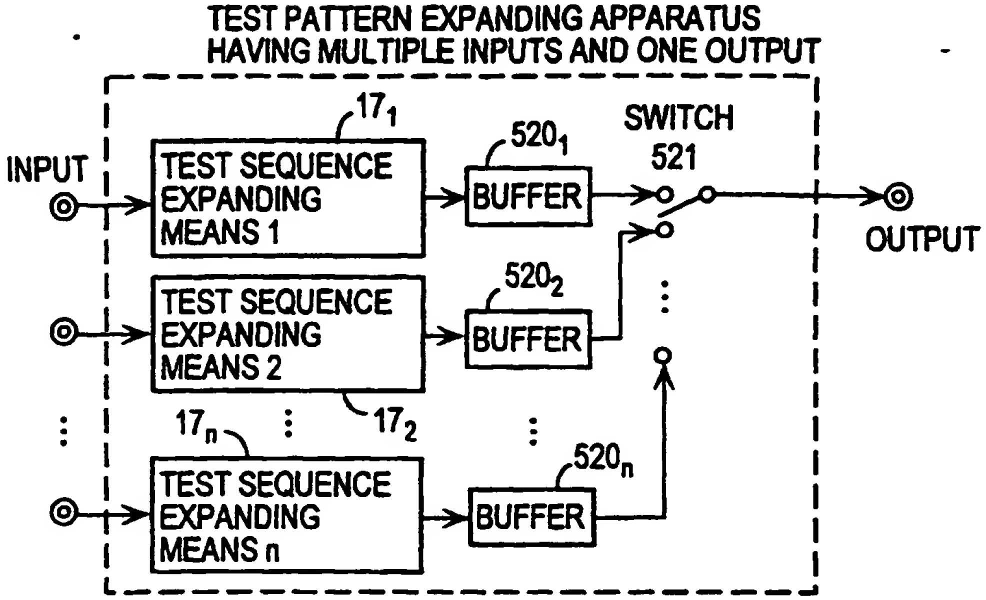

TEST PATTERN EXPANDING APPARATUS
HAVING MULTIPLE INPUTS AND ONE OUTPUT
INPUT
17_1
TEST SEQUENCE EXPANDING MEANS 1
520_1
BUFFER
SWITCH
521
OUTPUT
TEST SEQUENCE EXPANDING MEANS 2
520_2
BUFFER
17_2
17_n
TEST SEQUENCE EXPANDING MEANS n
520_n
BUFFER

FIG. 79

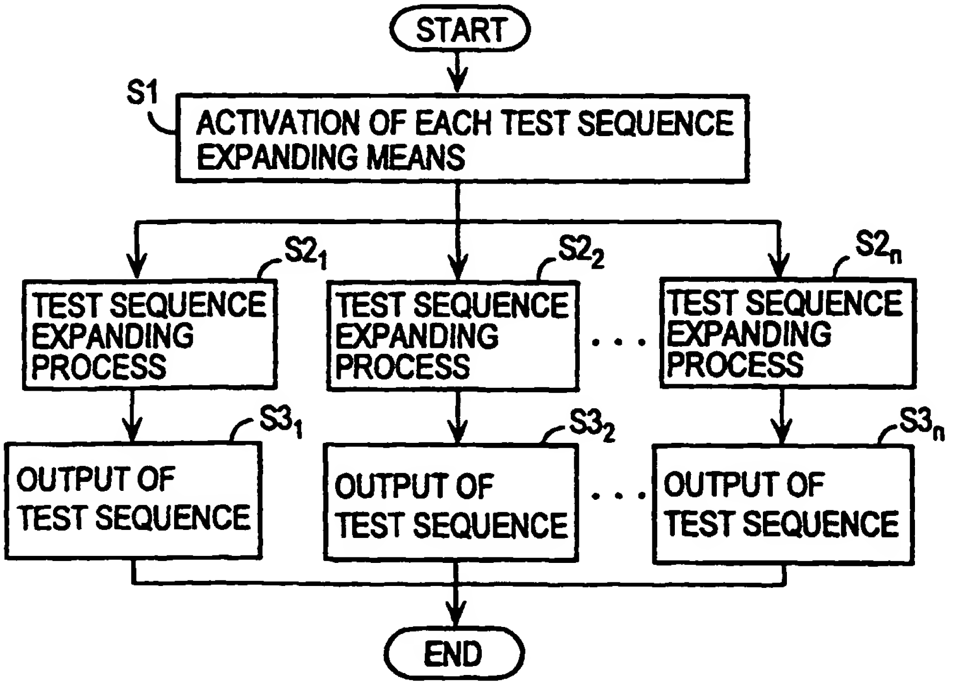

START
S1
ACTIVATION OF EACH TEST SEQUENCE EXPANDING MEANS
$S2_1$
$S2_2$
$S2_n$
TEST SEQUENCE EXPANDING PROCESS
TEST SEQUENCE EXPANDING PROCESS
TEST SEQUENCE EXPANDING PROCESS
$S3_1$
$S3_2$
$S3_n$
OUTPUT OF TEST SEQUENCE
OUTPUT OF TEST SEQUENCE
OUTPUT OF TEST SEQUENCE
END

FIG. 80

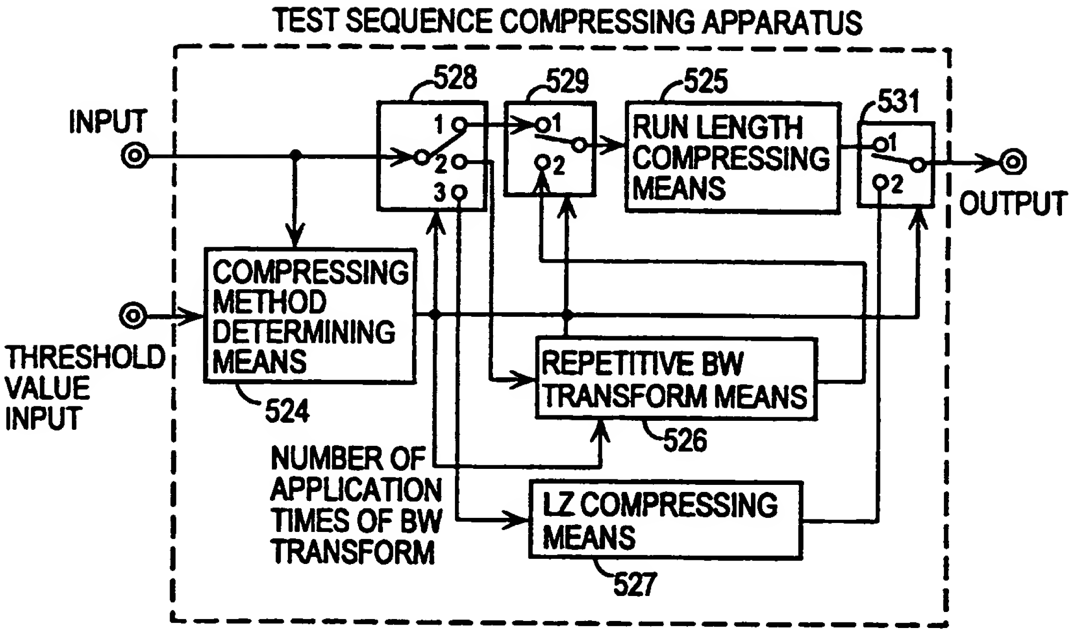

TEST SEQUENCE COMPRESSING APPARATUS
INPUT
THRESHOLD VALUE INPUT
RUN LENGTH COMPRESSING MEANS
OUTPUT
COMPRESSING METHOD DETERMINING MEANS
REPETITIVE BW TRANSFORM MEANS
NUMBER OF APPLICATION TIMES OF BW TRANSFORM
LZ COMPRESSING MEANS

FIG. 81

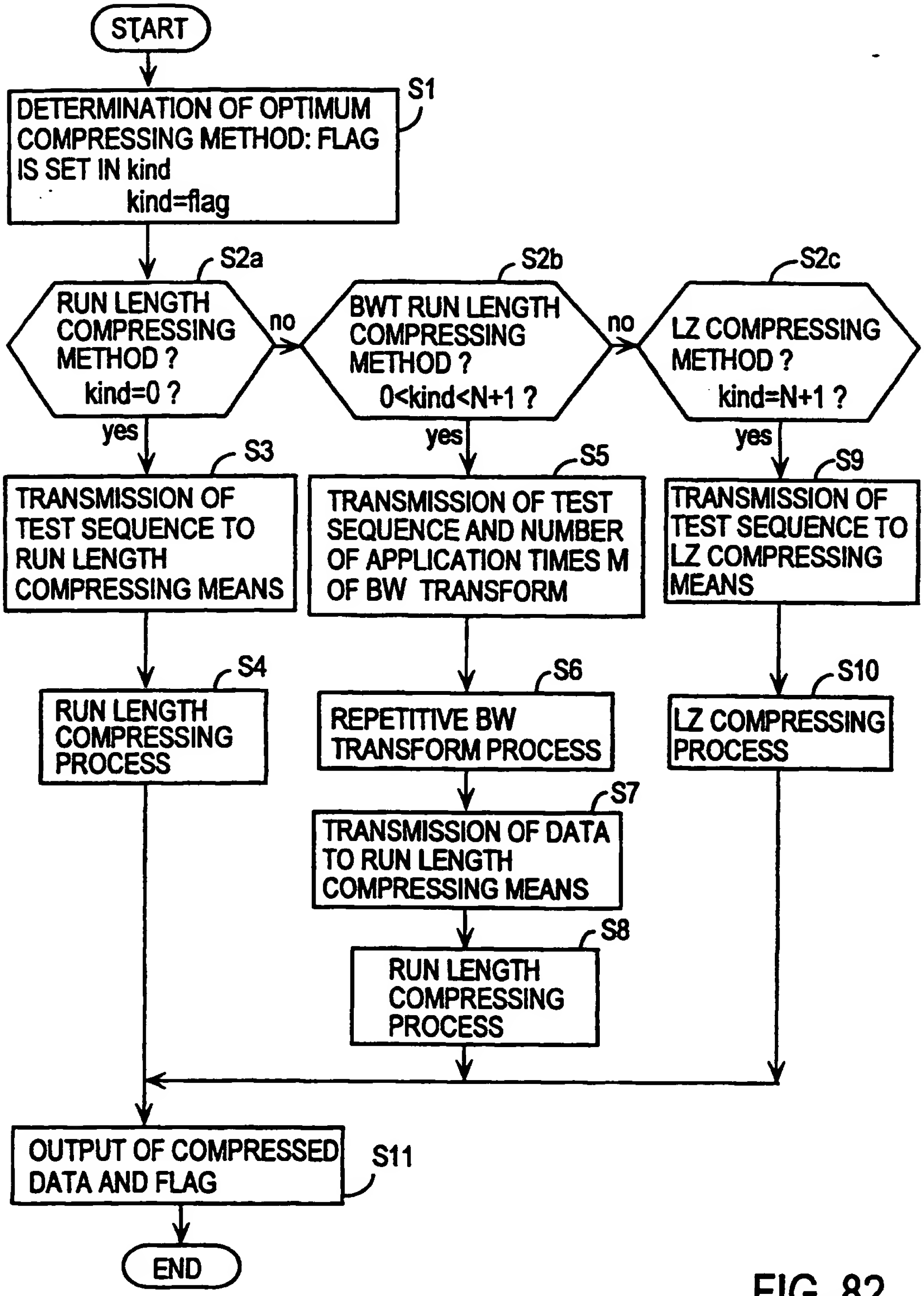

START
DETERMINATION OF OPTIMUM COMPRESSING METHOD: FLAG IS SET IN kind
kind=flag
S1
RUN LENGTH COMPRESSING METHOD ? kind=0 ?
S2a
no
BWT RUN LENGTH COMPRESSING METHOD ? 0<kind<N+1 ?
S2b
no
LZ COMPRESSING METHOD ? kind=N+1 ?
S2c
yes
yes
yes
TRANSMISSION OF TEST SEQUENCE TO RUN LENGTH COMPRESSING MEANS
S3
TRANSMISSION OF TEST SEQUENCE AND NUMBER OF APPLICATION TIMES M OF BW TRANSFORM
S5
TRANSMISSION OF TEST SEQUENCE TO LZ COMPRESSING MEANS
S9
RUN LENGTH COMPRESSING PROCESS
S4
REPETITIVE BW TRANSFORM PROCESS
S6
LZ COMPRESSING PROCESS
S10
TRANSMISSION OF DATA TO RUN LENGTH COMPRESSING MEANS
S7
RUN LENGTH COMPRESSING PROCESS
S8
OUTPUT OF COMPRESSED DATA AND FLAG
S11
END

FIG. 82

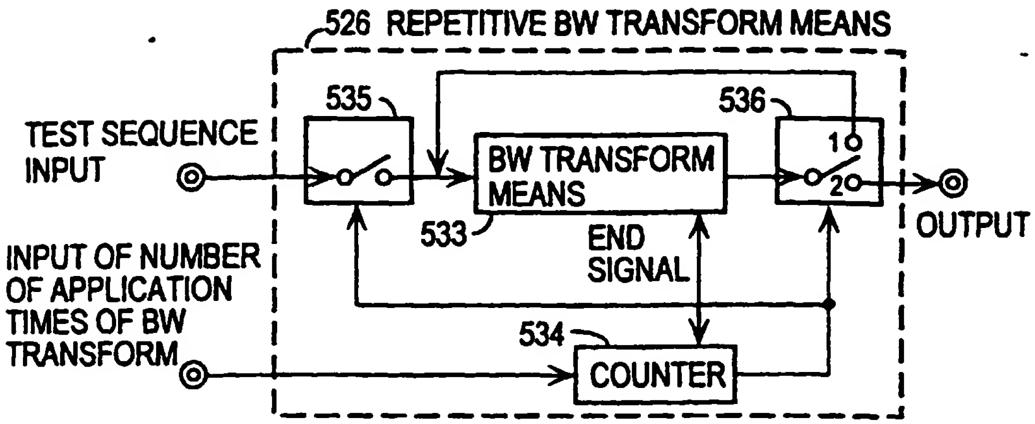

526 REPETITIVE BW TRANSFORM MEANS
535
536
TEST SEQUENCE INPUT
BW TRANSFORM MEANS
1
2
OUTPUT
533
END SIGNAL
INPUT OF NUMBER OF APPLICATION TIMES OF BW TRANSFORM
534
COUNTER

FIG. 83

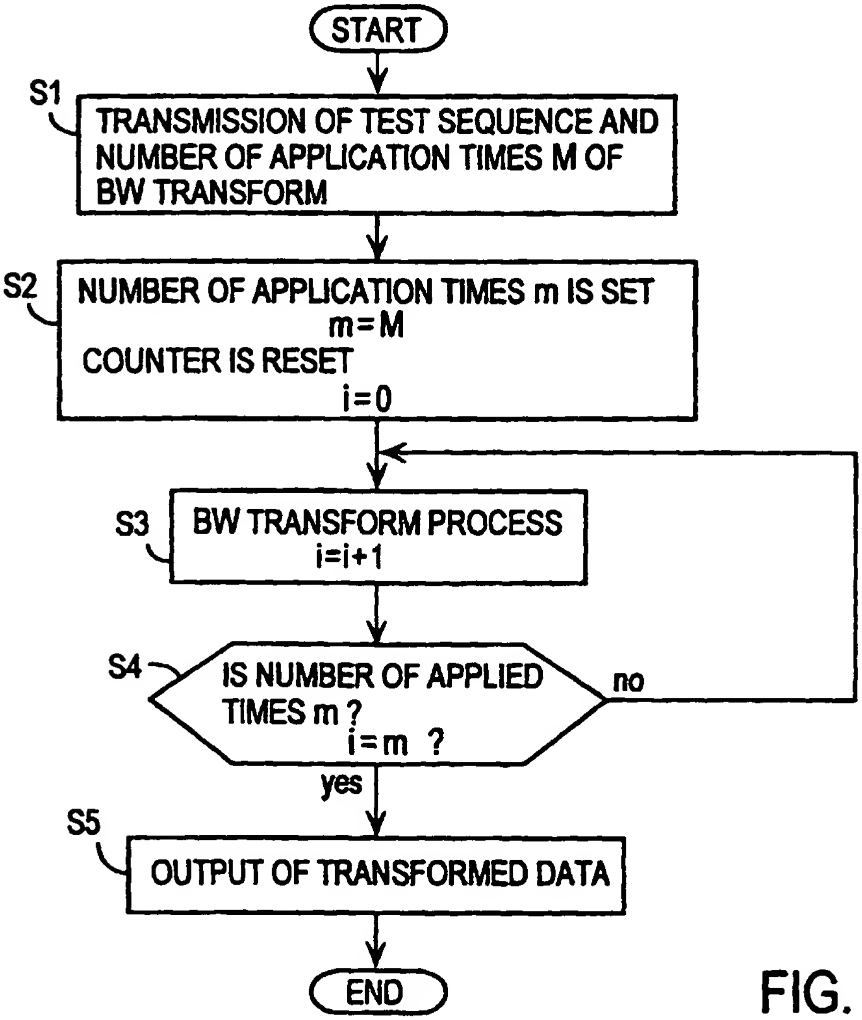

START
S1
TRANSMISSION OF TEST SEQUENCE AND NUMBER OF APPLICATION TIMES M OF BW TRANSFORM
S2
NUMBER OF APPLICATION TIMES m IS SET
m=M
COUNTER IS RESET
i=0
S3
BW TRANSFORM PROCESS
i=i+1
S4
IS NUMBER OF APPLIED TIMES m ?
i=m ?
no
yes
S5
OUTPUT OF TRANSFORMED DATA
END

FIG. 84

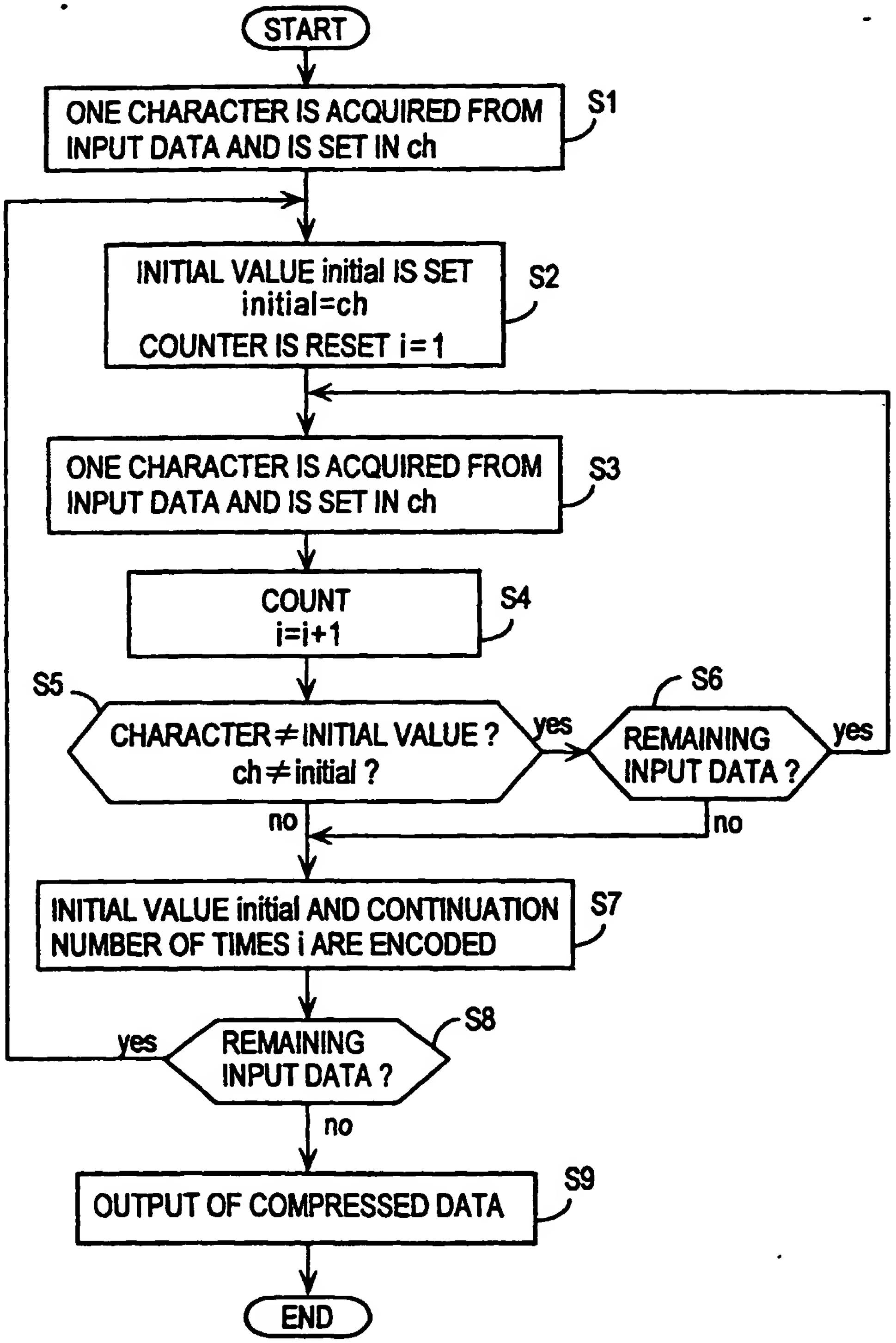

START
ONE CHARACTER IS ACQUIRED FROM INPUT DATA AND IS SET IN ch
S1
INITIAL VALUE initial IS SET
initial=ch
COUNTER IS RESET i=1
S2
ONE CHARACTER IS ACQUIRED FROM INPUT DATA AND IS SET IN ch
S3
COUNT
i=i+1
S4
S5
CHARACTER≠INITIAL VALUE ?
ch≠initial ?
yes
S6
REMAINING INPUT DATA ?
yes
no
no
INITIAL VALUE initial AND CONTINUATION NUMBER OF TIMES i ARE ENCODED
S7
yes
REMAINING INPUT DATA ?
S8
no
OUTPUT OF COMPRESSED DATA
S9
END

FIG.85

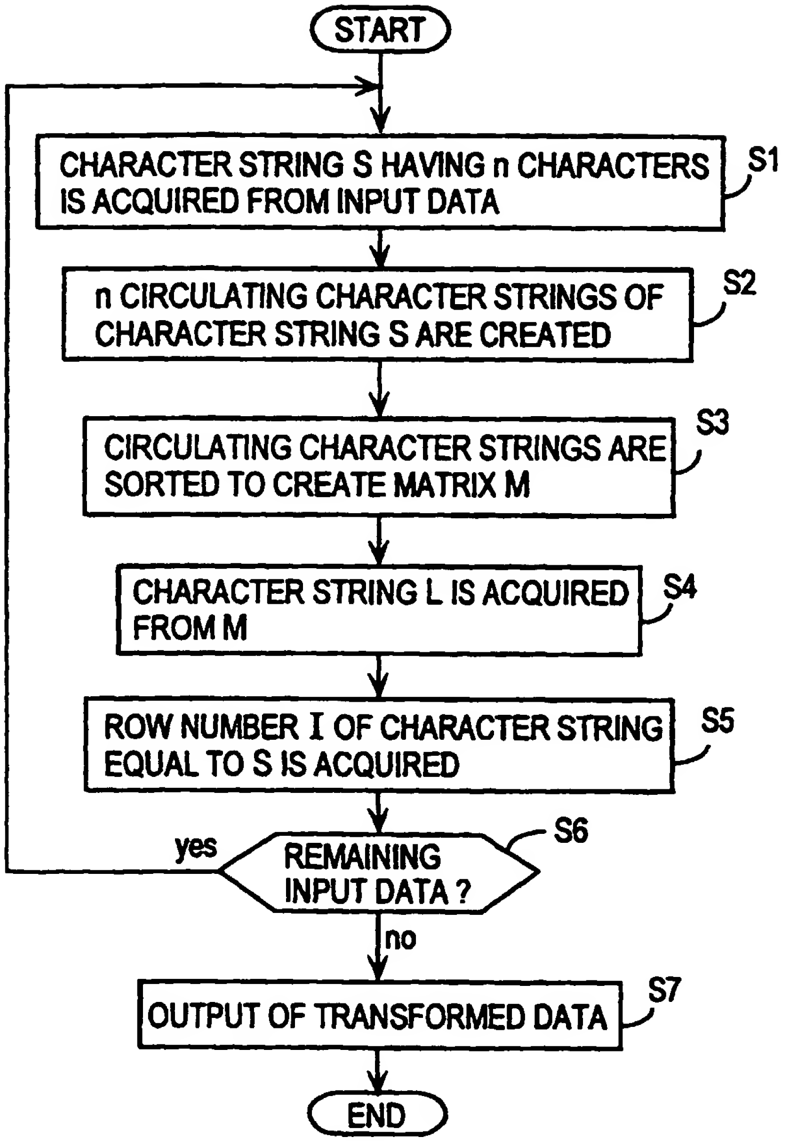

START
CHARACTER STRING S HAVING n CHARACTERS IS ACQUIRED FROM INPUT DATA
S1
n CIRCULATING CHARACTER STRINGS OF CHARACTER STRING S ARE CREATED
S2
CIRCULATING CHARACTER STRINGS ARE SORTED TO CREATE MATRIX M
S3
CHARACTER STRING L IS ACQUIRED FROM M
S4
ROW NUMBER I OF CHARACTER STRING EQUAL TO S IS ACQUIRED
S5
REMAINING INPUT DATA ?
S6
yes
no
OUTPUT OF TRANSFORMED DATA
S7
END

FIG. 86

FIG. 87

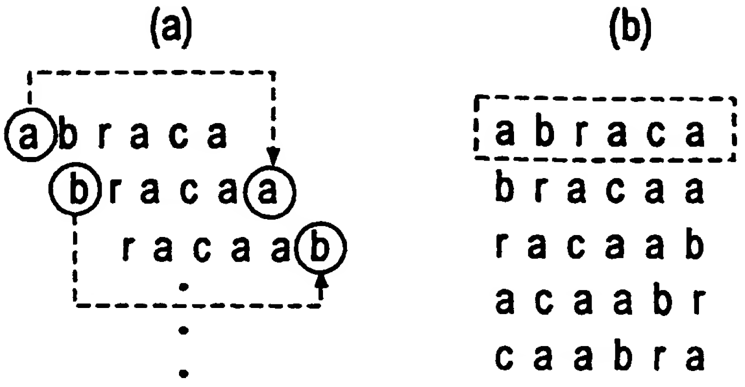

(a)
abraca
bracaa
racaab
(b)
abraca
bracaa
racaab
acaabr
caabra
aabrac

FIG. 88

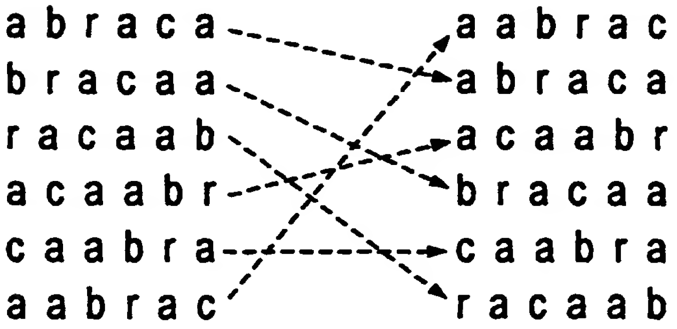
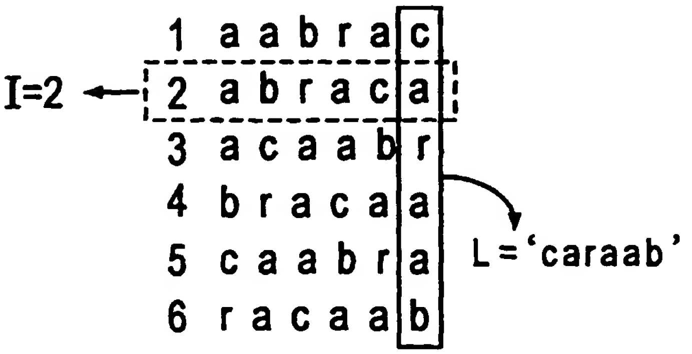

abraca
bracaa
racaab
acaabr
caabra
aabrac
aabrac
abraca
acaabr
bracaa
caabra
racaab

FIG. 89


1 aabrac
I=2
2 abraca
3 acaabr
4 bracaa
5 caabra
6 racaab
L='caraab'

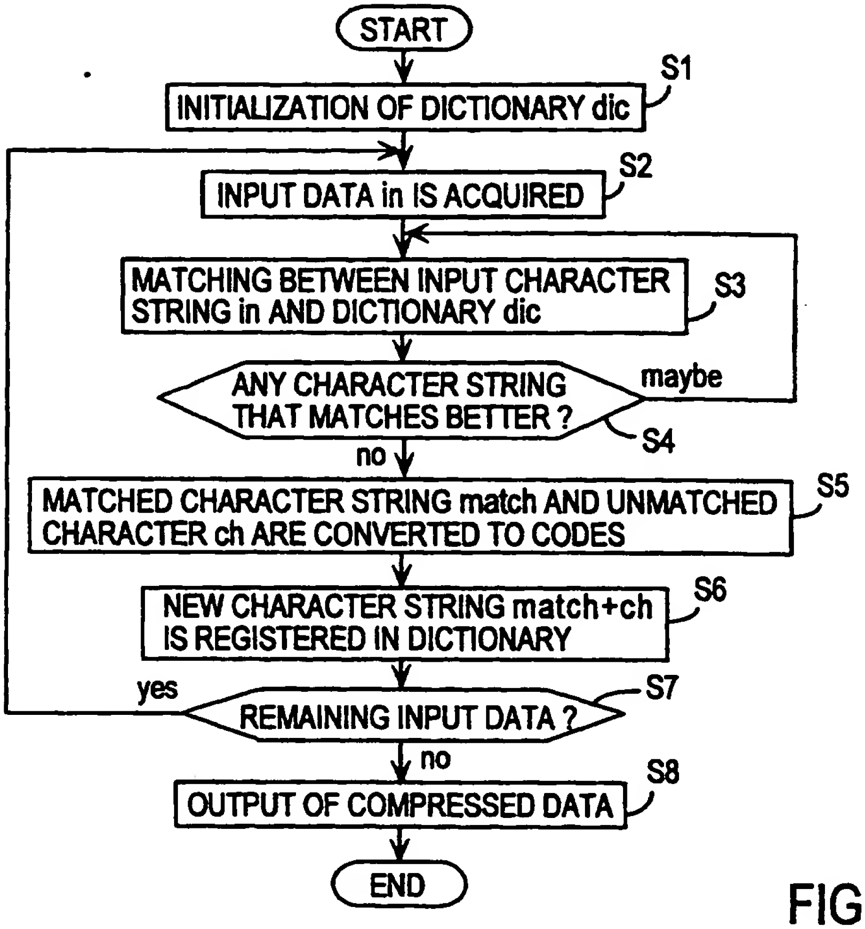

START
INITIALIZATION OF DICTIONARY dic
S1
INPUT DATA in IS ACQUIRED
S2
MATCHING BETWEEN INPUT CHARACTER STRING in AND DICTIONARY dic
S3
ANY CHARACTER STRING THAT MATCHES BETTER ?
maybe
S4
no
MATCHED CHARACTER STRING match AND UNMATCHED CHARACTER ch ARE CONVERTED TO CODES
S5
NEW CHARACTER STRING match+ch IS REGISTERED IN DICTIONARY
S6
yes
REMAINING INPUT DATA ?
S7
no
OUTPUT OF COMPRESSED DATA
S8
END

FIG. 90

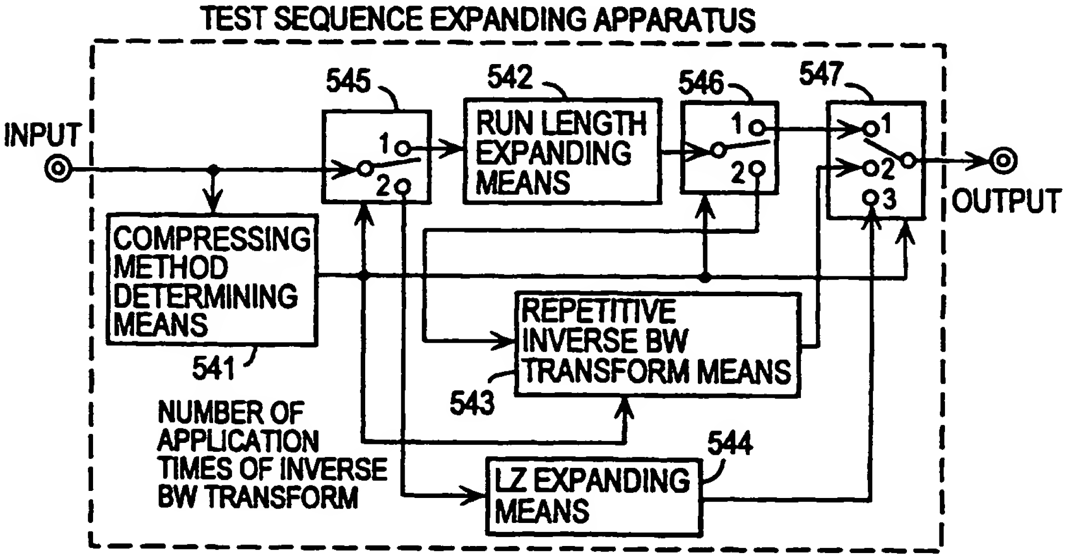

TEST SEQUENCE EXPANDING APPARATUS
INPUT
RUN LENGTH EXPANDING MEANS
OUTPUT
COMPRESSING METHOD DETERMINING MEANS
REPETITIVE INVERSE BW TRANSFORM MEANS
NUMBER OF APPLICATION TIMES OF INVERSE BW TRANSFORM
LZ EXPANDING MEANS

FIG. 91

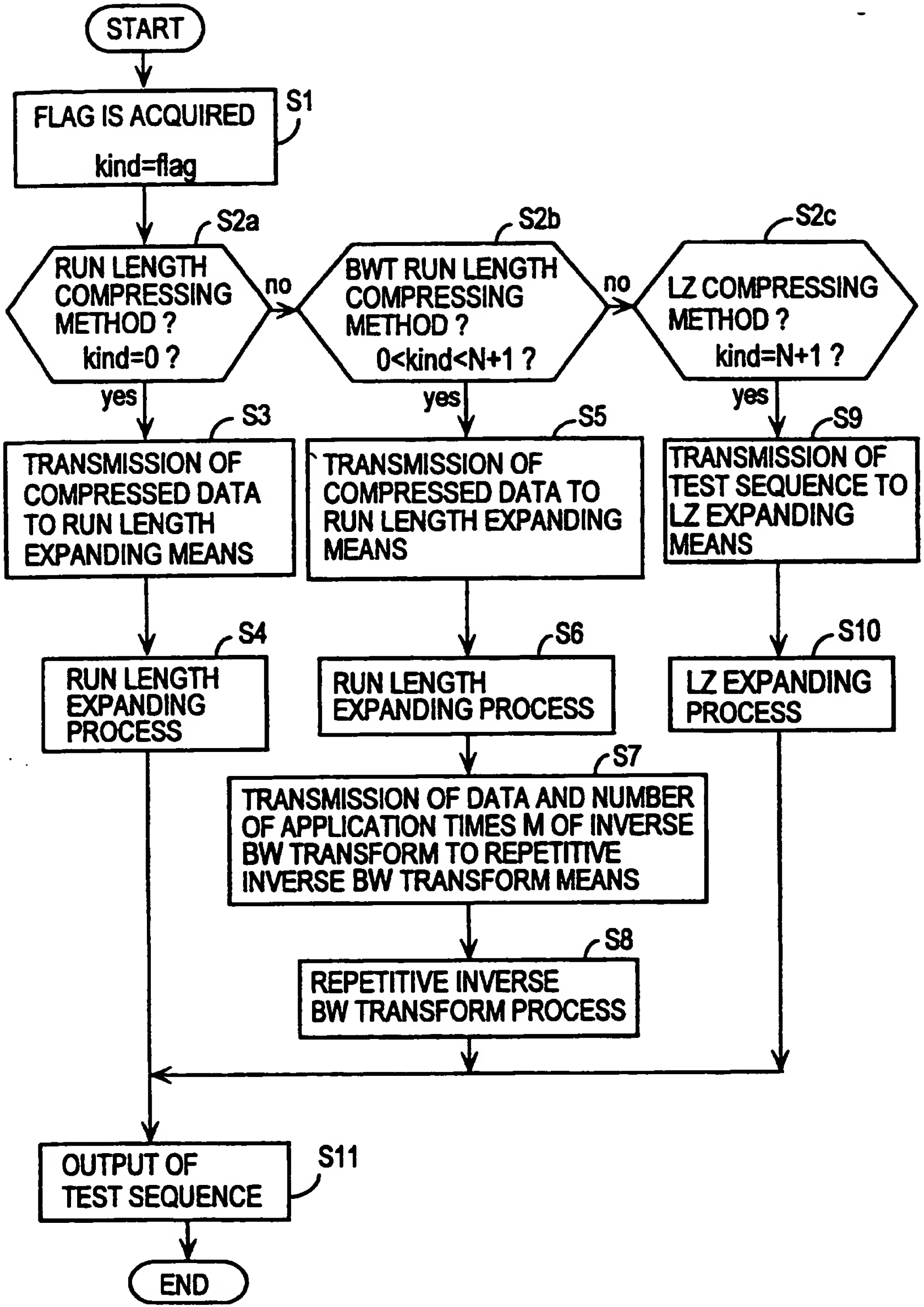

START
FLAG IS ACQUIRED
kind=flag
S1
S2a
RUN LENGTH COMPRESSING METHOD ?
kind=0 ?
no
S2b
BWT RUN LENGTH COMPRESSING METHOD ?
0<kind<N+1 ?
no
S2c
LZ COMPRESSING METHOD ?
kind=N+1 ?
yes
S3
TRANSMISSION OF COMPRESSED DATA TO RUN LENGTH EXPANDING MEANS
yes
S5
TRANSMISSION OF COMPRESSED DATA TO RUN LENGTH EXPANDING MEANS
yes
S9
TRANSMISSION OF TEST SEQUENCE TO LZ EXPANDING MEANS
S4
RUN LENGTH EXPANDING PROCESS
S6
RUN LENGTH EXPANDING PROCESS
S10
LZ EXPANDING PROCESS
S7
TRANSMISSION OF DATA AND NUMBER OF APPLICATION TIMES M OF INVERSE BW TRANSFORM TO REPETITIVE INVERSE BW TRANSFORM MEANS
S8
REPETITIVE INVERSE BW TRANSFORM PROCESS
OUTPUT OF TEST SEQUENCE
S11
END

FIG. 92

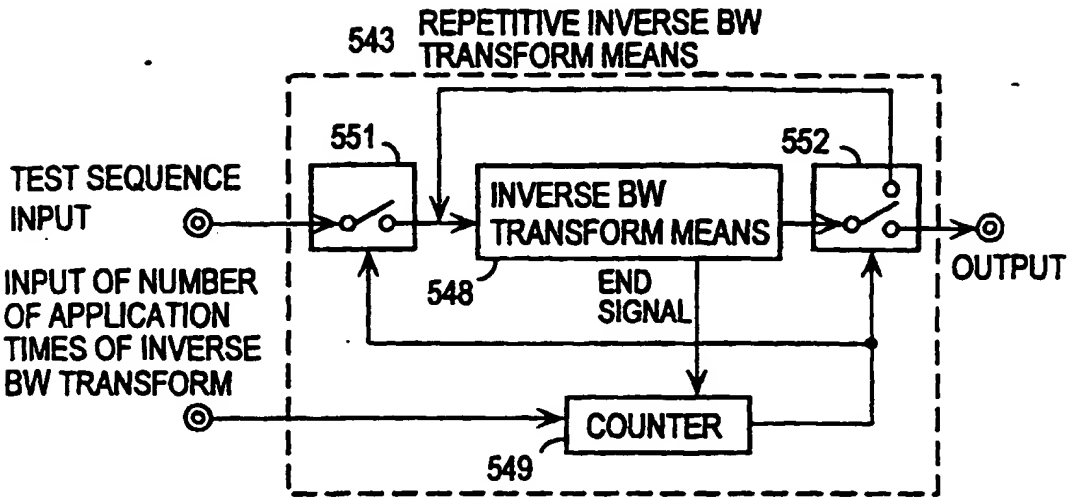

543 REPETITIVE INVERSE BW TRANSFORM MEANS
551
552
TEST SEQUENCE INPUT
INVERSE BW TRANSFORM MEANS
OUTPUT
INPUT OF NUMBER OF APPLICATION TIMES OF INVERSE BW TRANSFORM
548
END SIGNAL
COUNTER
549

FIG. 93

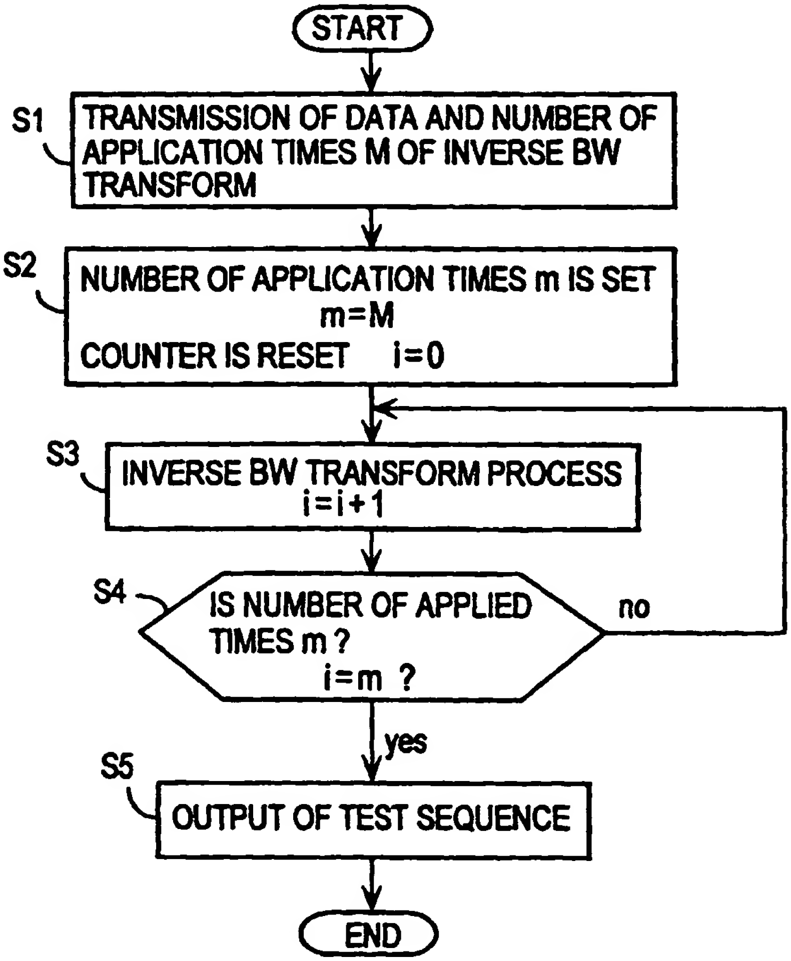

START
S1
TRANSMISSION OF DATA AND NUMBER OF APPLICATION TIMES M OF INVERSE BW TRANSFORM
S2
NUMBER OF APPLICATION TIMES m IS SET m=M
COUNTER IS RESET i=0
S3
INVERSE BW TRANSFORM PROCESS i=i+1
S4
IS NUMBER OF APPLIED TIMES m ? i=m ?
no
yes
S5
OUTPUT OF TEST SEQUENCE
END

FIG. 94

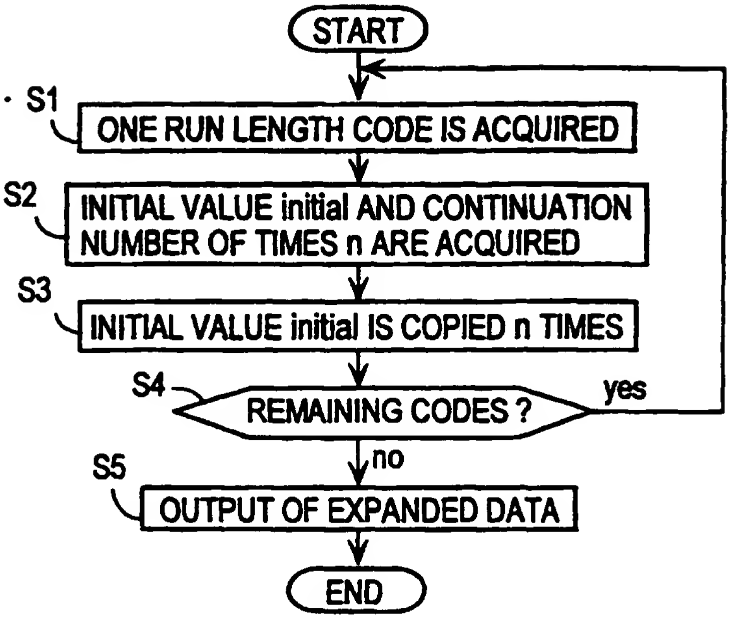

START
S1
ONE RUN LENGTH CODE IS ACQUIRED
S2
INITIAL VALUE initial AND CONTINUATION NUMBER OF TIMES n ARE ACQUIRED
S3
INITIAL VALUE initial IS COPIED n TIMES
S4
REMAINING CODES ?
yes
no
S5
OUTPUT OF EXPANDED DATA
END

FIG. 95

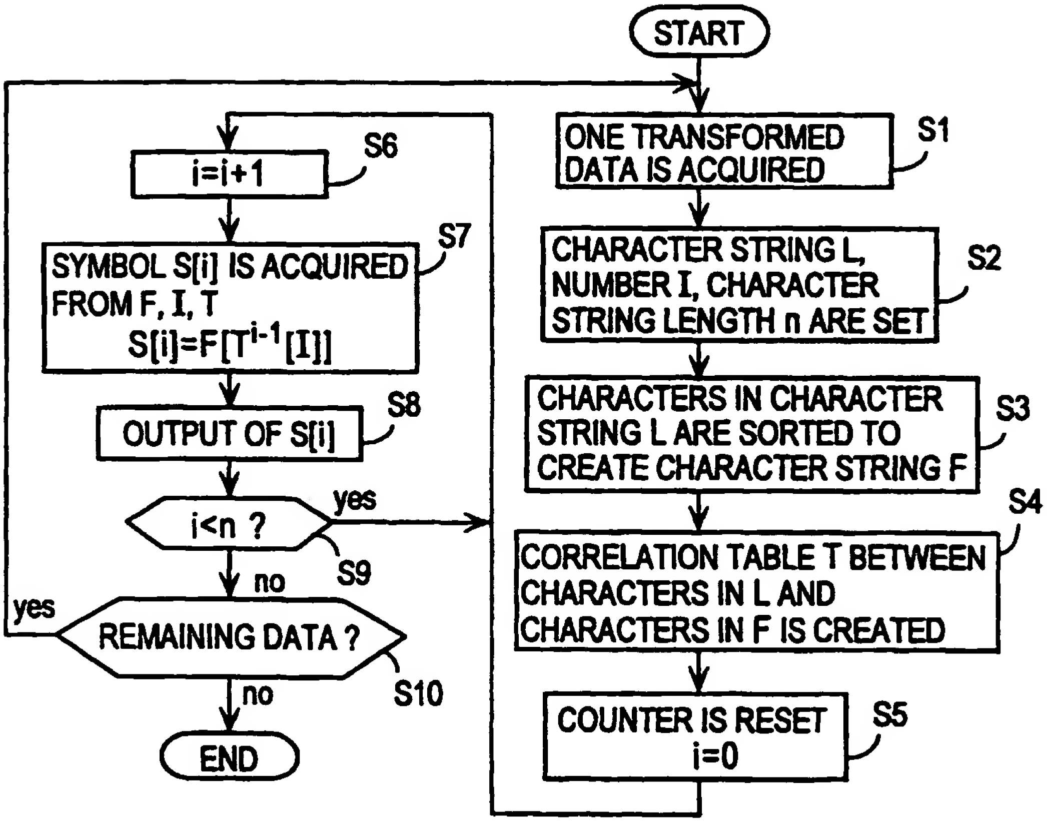

START
ONE TRANSFORMED DATA IS ACQUIRED
S1
CHARACTER STRING L, NUMBER I, CHARACTER STRING LENGTH n ARE SET
S2
CHARACTERS IN CHARACTER STRING L ARE SORTED TO CREATE CHARACTER STRING F
S3
S4
CORRELATION TABLE T BETWEEN CHARACTERS IN L AND CHARACTERS IN F IS CREATED
COUNTER IS RESET i=0
S5
i=i+1
S6
SYMBOL S[i] IS ACQUIRED FROM F, I, T S[i]=F[T^{i-1}[I]]
S7
OUTPUT OF S[i]
S8
i<n ?
yes
S9
no
yes
REMAINING DATA ?
S10
no
END

FIG. 96

FIG. 97

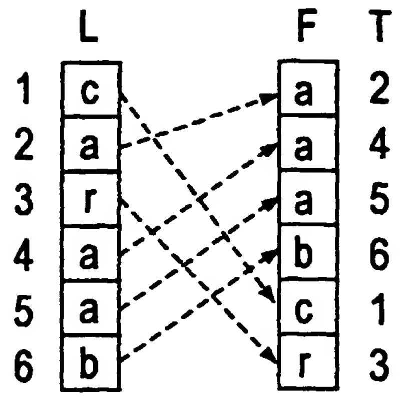

L
F
T
1
2
3
4
5
6
c
a
r
a
a
b
a
a
a
b
c
r
2
4
5
6
1
3

FIG. 98

i	$T^{i-1}[I]$	$S[i]=F[T^{i-1}[I]]$
1	2	a
2	4	b
3	6	r
4	3	a
5	5	c
6	1	a

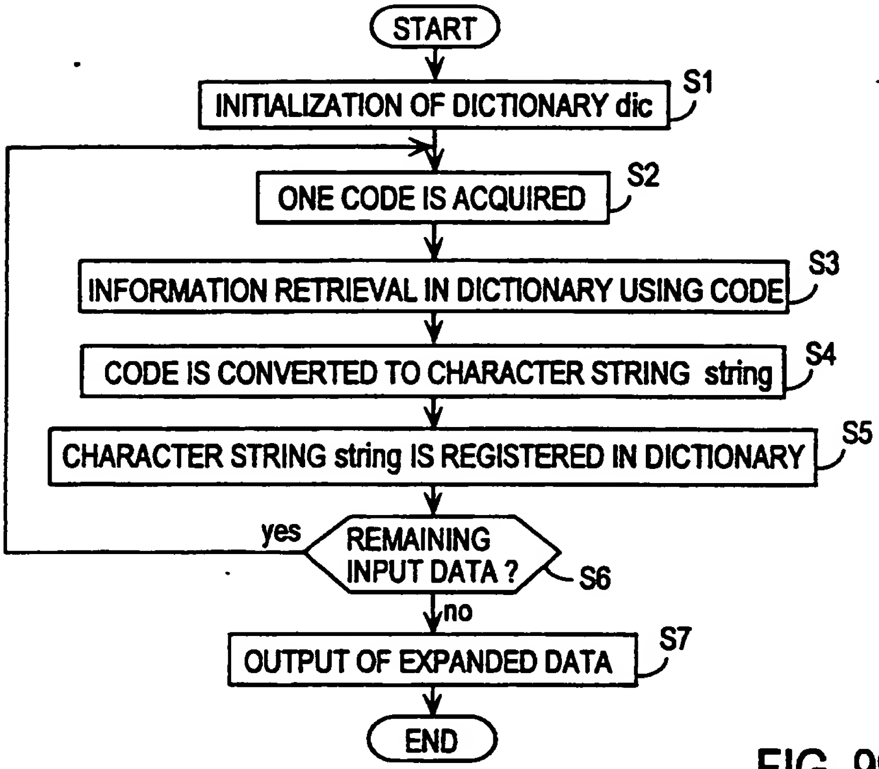

START
INITIALIZATION OF DICTIONARY dic
S1
ONE CODE IS ACQUIRED
S2
INFORMATION RETRIEVAL IN DICTIONARY USING CODE
S3
CODE IS CONVERTED TO CHARACTER STRING string
S4
CHARACTER STRING string IS REGISTERED IN DICTIONARY
S5
yes
REMAINING INPUT DATA ?
S6
no
OUTPUT OF EXPANDED DATA
S7
END

FIG. 99

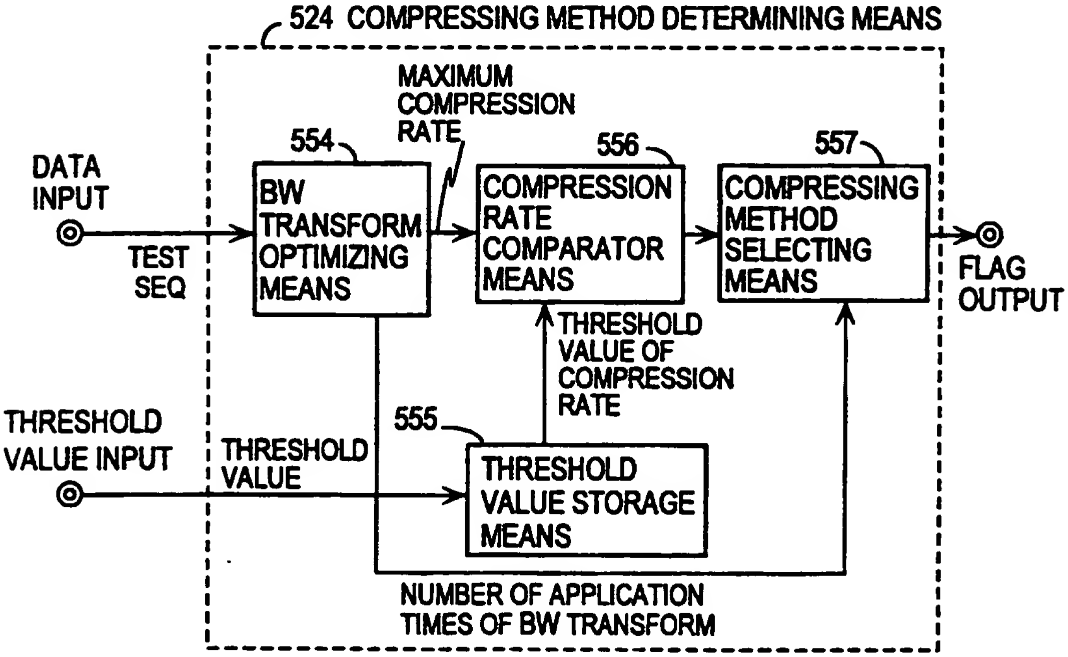

524 COMPRESSING METHOD DETERMINING MEANS
DATA INPUT
TEST SEQ
554
BW TRANSFORM OPTIMIZING MEANS
MAXIMUM COMPRESSION RATE
556
COMPRESSION RATE COMPARATOR MEANS
557
COMPRESSING METHOD SELECTING MEANS
FLAG OUTPUT
THRESHOLD VALUE OF COMPRESSION RATE
555
THRESHOLD VALUE STORAGE MEANS
THRESHOLD VALUE INPUT
THRESHOLD VALUE
NUMBER OF APPLICATION TIMES OF BW TRANSFORM

FIG. 100

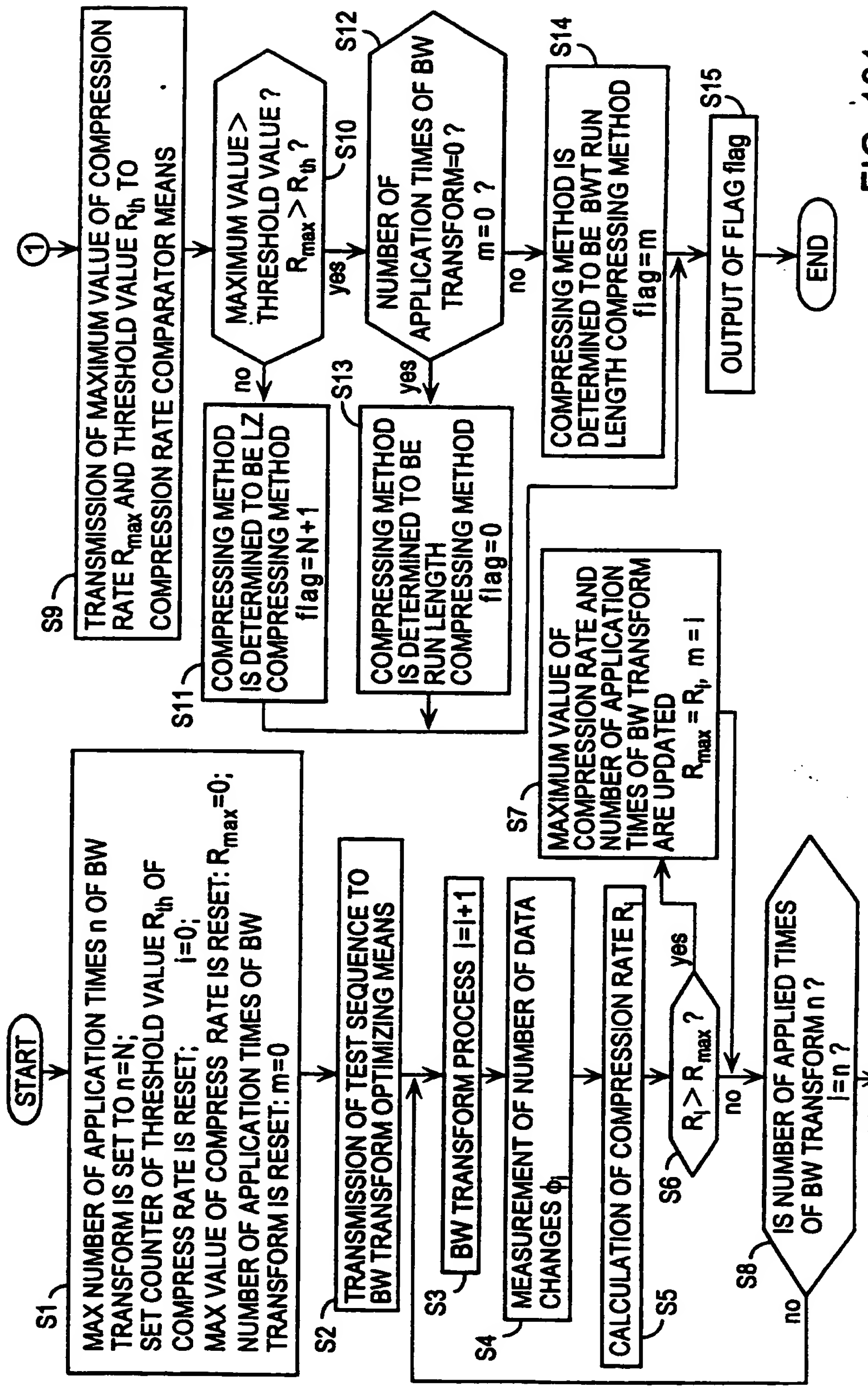

START
S1
MAX NUMBER OF APPLICATION TIMES n OF BW TRANSFORM IS SET TO n=N;
SET COUNTER OF THRESHOLD VALUE R_{th} OF COMPRESS RATE IS RESET; l=0;
MAX VALUE OF COMPRESS RATE IS RESET: R_{max}=0;
NUMBER OF APPLICATION TIMES OF BW TRANSFORM IS RESET: m=0
S2
TRANSMISSION OF TEST SEQUENCE TO BW TRANSFORM OPTIMIZING MEANS
S3
BW TRANSFORM PROCESS l=l+1
S4
MEASUREMENT OF NUMBER OF DATA CHANGES ϕ_l
S5
CALCULATION OF COMPRESSION RATE R_l
S6
$R_l > R_{max}$?
yes
no
S7
MAXIMUM VALUE OF COMPRESSION RATE AND NUMBER OF APPLICATION TIMES OF BW TRANSFORM ARE UPDATED
$R_{max} = R_l$, m = l
S8
IS NUMBER OF APPLIED TIMES OF BW TRANSFORM n ?
l=n ?
yes
no
①
①
S9
TRANSMISSION OF MAXIMUM VALUE OF COMPRESSION RATE R_{max} AND THRESHOLD VALUE R_{th} TO COMPRESSION RATE COMPARATOR MEANS
S10
MAXIMUM VALUE > THRESHOLD VALUE ?
$R_{max} > R_{th}$?
yes
no
S11
COMPRESSING METHOD IS DETERMINED TO BE LZ COMPRESSING METHOD
flag=N+1
S12
NUMBER OF APPLICATION TIMES OF BW TRANSFORM=0 ?
m=0 ?
yes
no
S13
COMPRESSING METHOD IS DETERMINED TO BE RUN LENGTH COMPRESSING METHOD
flag=0
S14
COMPRESSING METHOD IS DETERMINED TO BE BWT RUN LENGTH COMPRESSING METHOD
flag=m
S15
OUTPUT OF FLAG flag
END

FIG. 101

FIG. 102

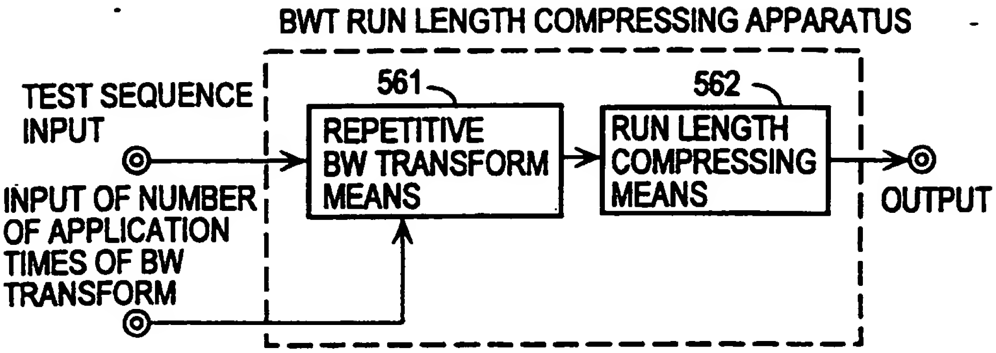

BWT RUN LENGTH COMPRESSING APPARATUS
TEST SEQUENCE INPUT
INPUT OF NUMBER OF APPLICATION TIMES OF BW TRANSFORM
561
REPETITIVE BW TRANSFORM MEANS
562
RUN LENGTH COMPRESSING MEANS
OUTPUT

FIG. 103

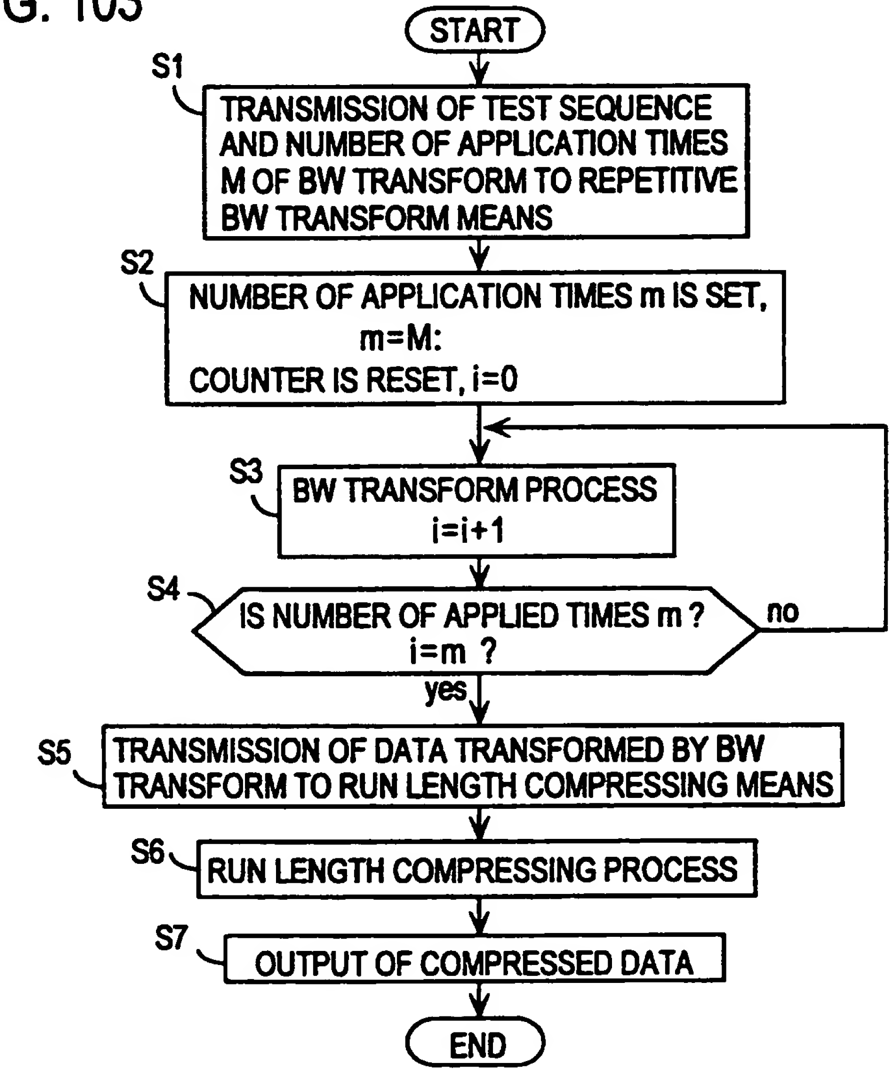

START
S1
TRANSMISSION OF TEST SEQUENCE AND NUMBER OF APPLICATION TIMES M OF BW TRANSFORM TO REPETITIVE BW TRANSFORM MEANS
S2
NUMBER OF APPLICATION TIMES m IS SET, m=M: COUNTER IS RESET, i=0
S3
BW TRANSFORM PROCESS i=i+1
S4
IS NUMBER OF APPLIED TIMES m ? i=m ?
no
yes
S5
TRANSMISSION OF DATA TRANSFORMED BY BW TRANSFORM TO RUN LENGTH COMPRESSING MEANS
S6
RUN LENGTH COMPRESSING PROCESS
S7
OUTPUT OF COMPRESSED DATA
END

FIG. 104

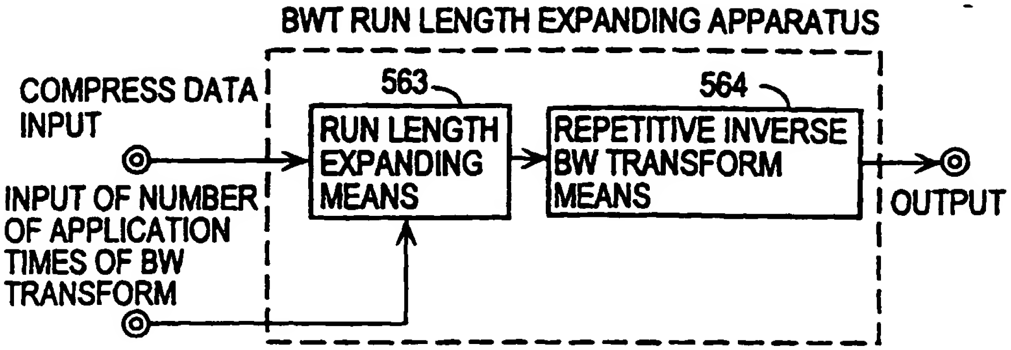
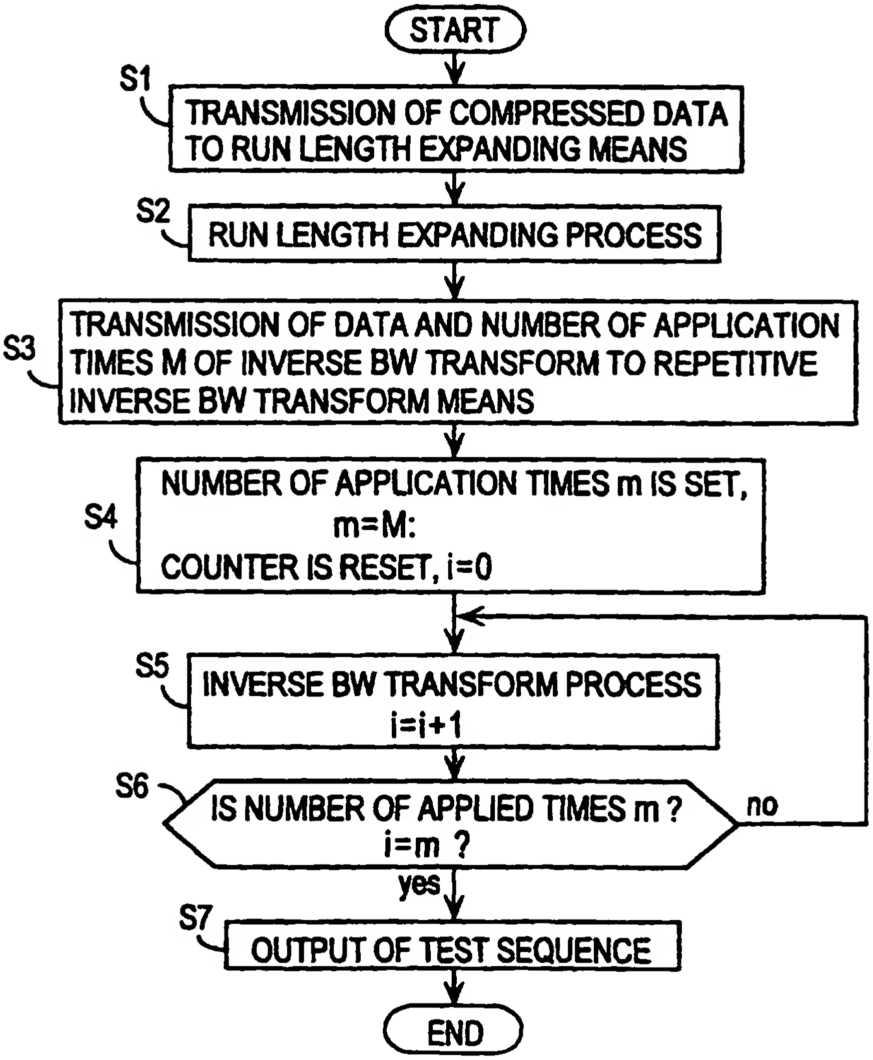

BWT RUN LENGTH EXPANDING APPARATUS
COMPRESS DATA INPUT
INPUT OF NUMBER OF APPLICATION TIMES OF BW TRANSFORM
563
RUN LENGTH EXPANDING MEANS
564
REPETITIVE INVERSE BW TRANSFORM MEANS
OUTPUT

FIG. 105


START
S1
TRANSMISSION OF COMPRESSED DATA TO RUN LENGTH EXPANDING MEANS
S2
RUN LENGTH EXPANDING PROCESS
S3
TRANSMISSION OF DATA AND NUMBER OF APPLICATION TIMES M OF INVERSE BW TRANSFORM TO REPETITIVE INVERSE BW TRANSFORM MEANS
S4
NUMBER OF APPLICATION TIMES m IS SET, m=M:
COUNTER IS RESET, i=0
S5
INVERSE BW TRANSFORM PROCESS i=i+1
S6
IS NUMBER OF APPLIED TIMES m ? i=m ?
no
yes
S7
OUTPUT OF TEST SEQUENCE
END

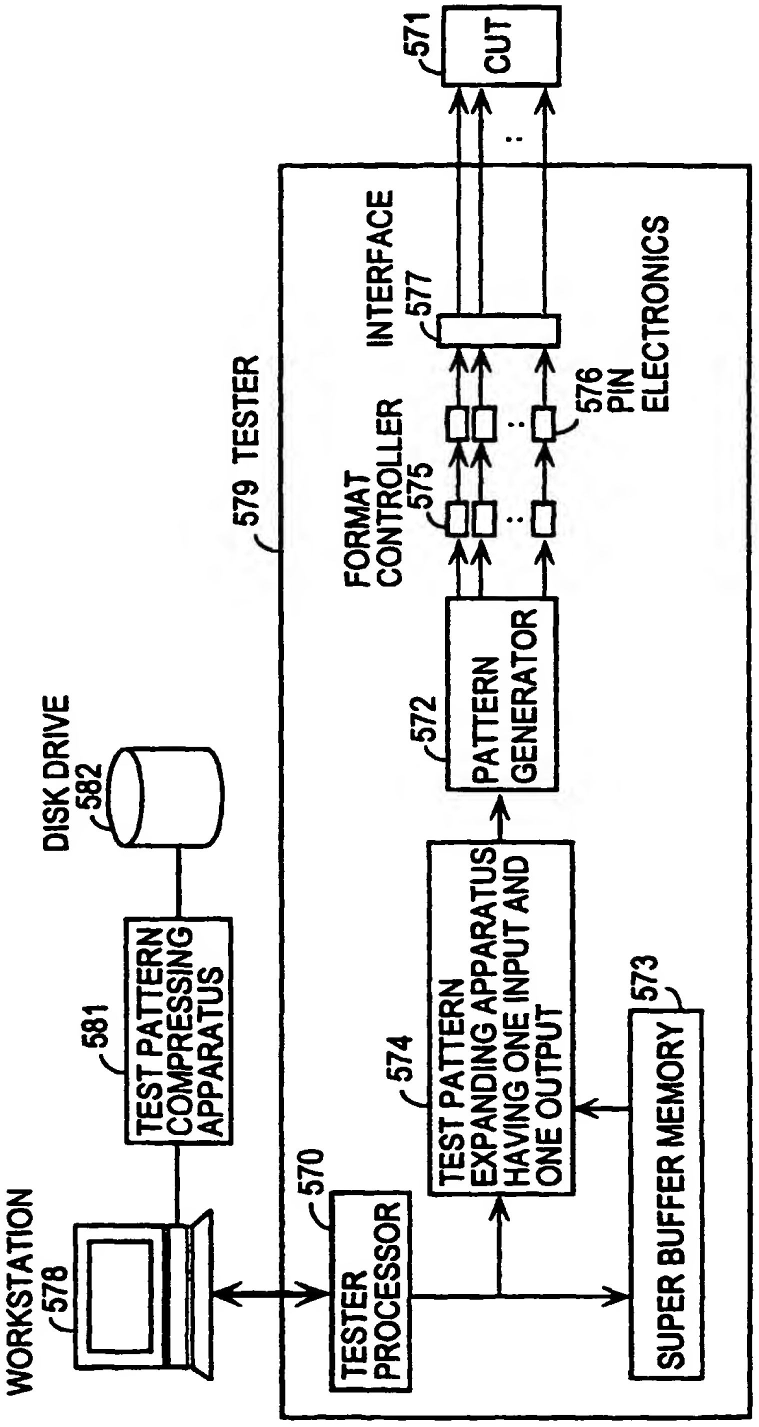

WORKSTATION
578
TEST PATTERN COMPRESSING APPARATUS
581
DISK DRIVE
582
579 TESTER
TESTER PROCESSOR
570
TEST PATTERN EXPANDING APPARATUS HAVING ONE INPUT AND ONE OUTPUT
574
SUPER BUFFER MEMORY
573
PATTERN GENERATOR
572
FORMAT CONTROLLER
575
576
PIN ELECTRONICS
INTERFACE
577
CUT
571

FIG. 106

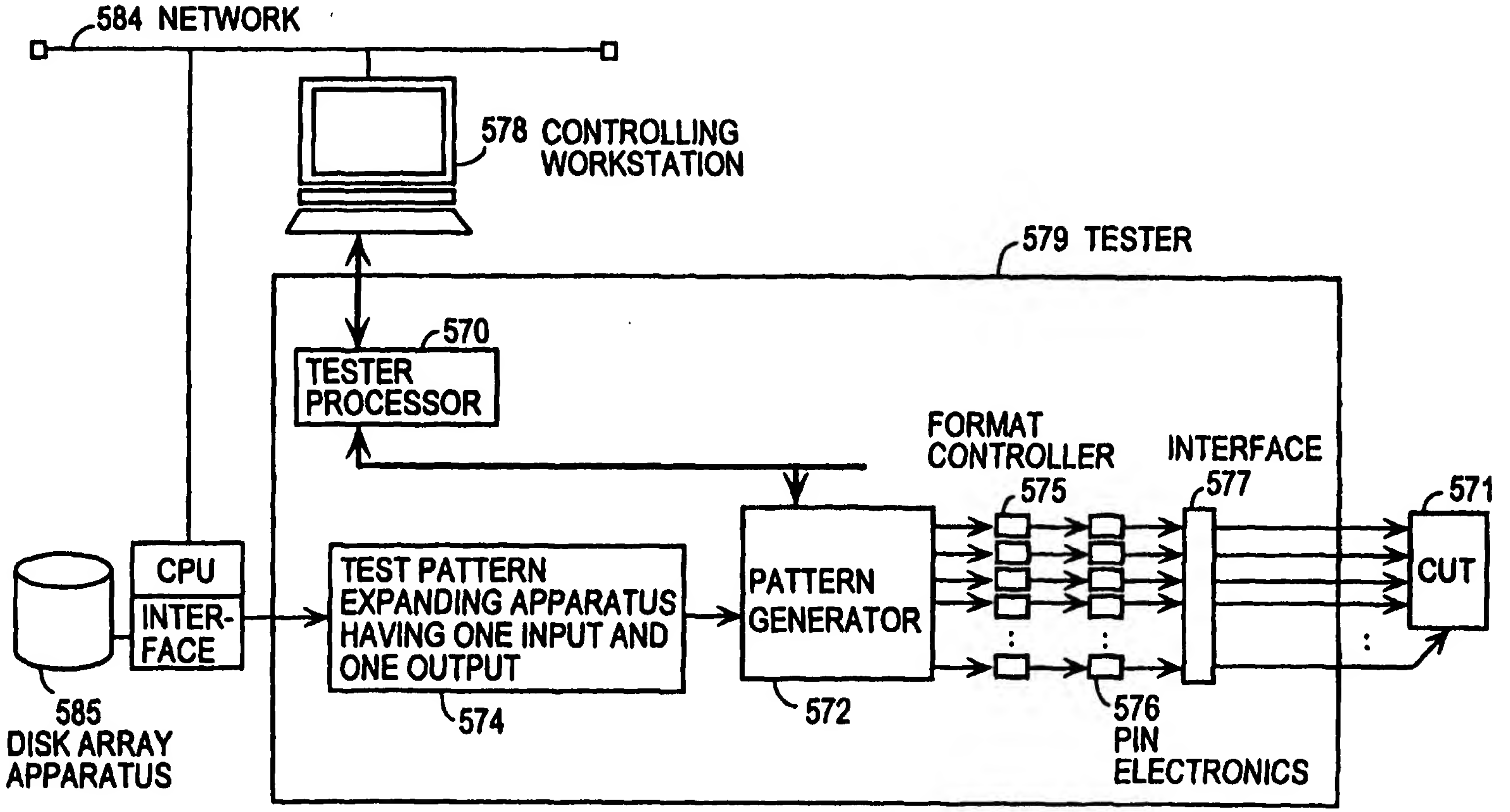

584 NETWORK
578 CONTROLLING WORKSTATION
579 TESTER
570
TESTER PROCESSOR
FORMAT CONTROLLER
575
INTERFACE
577
571
CUT
CPU
INTER-FACE
TEST PATTERN EXPANDING APPARATUS HAVING ONE INPUT AND ONE OUTPUT
PATTERN GENERATOR
585
DISK ARRAY APPARATUS
574
572
576
PIN ELECTRONICS

FIG. 107

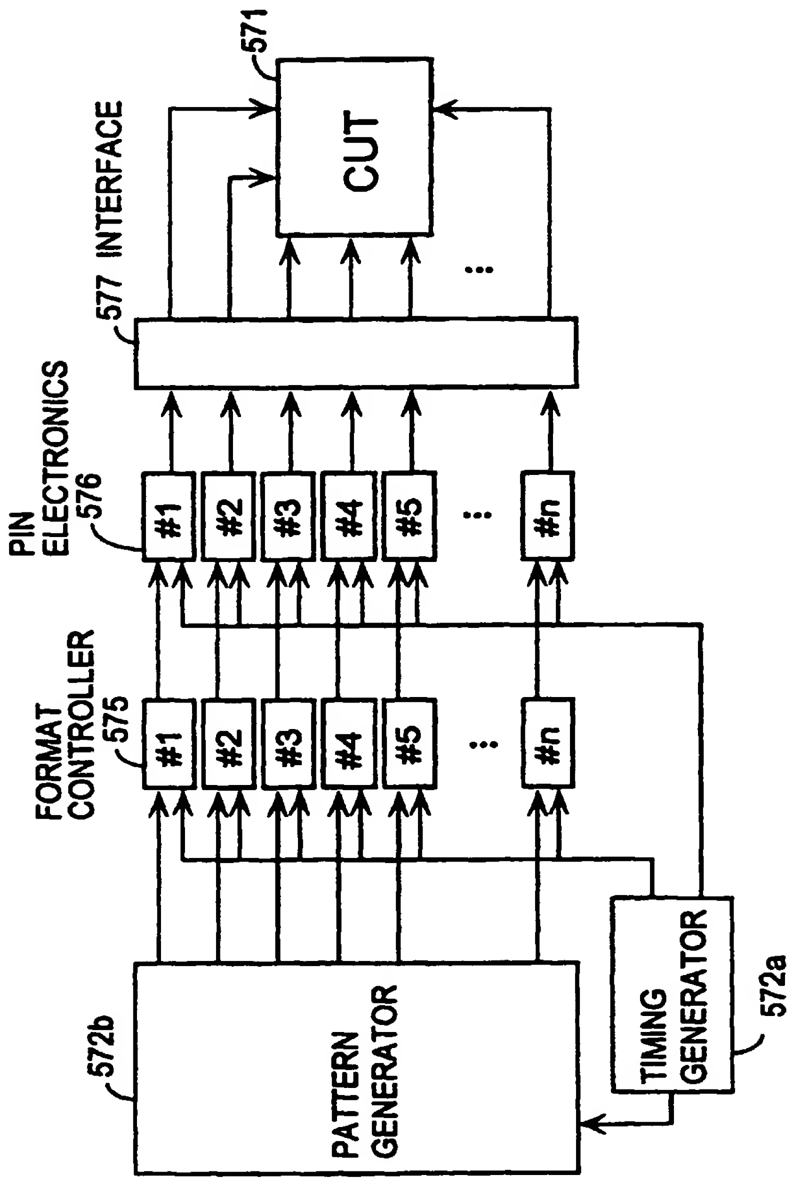

PATTERN GENERATOR
572b
TIMING GENERATOR
572a
FORMAT CONTROLLER
575
#1
#2
#3
#4
#5
...
#n
PIN ELECTRONICS
576
#1
#2
#3
#4
#5
...
#n
577 INTERFACE
CUT
571
...

FIG. 108

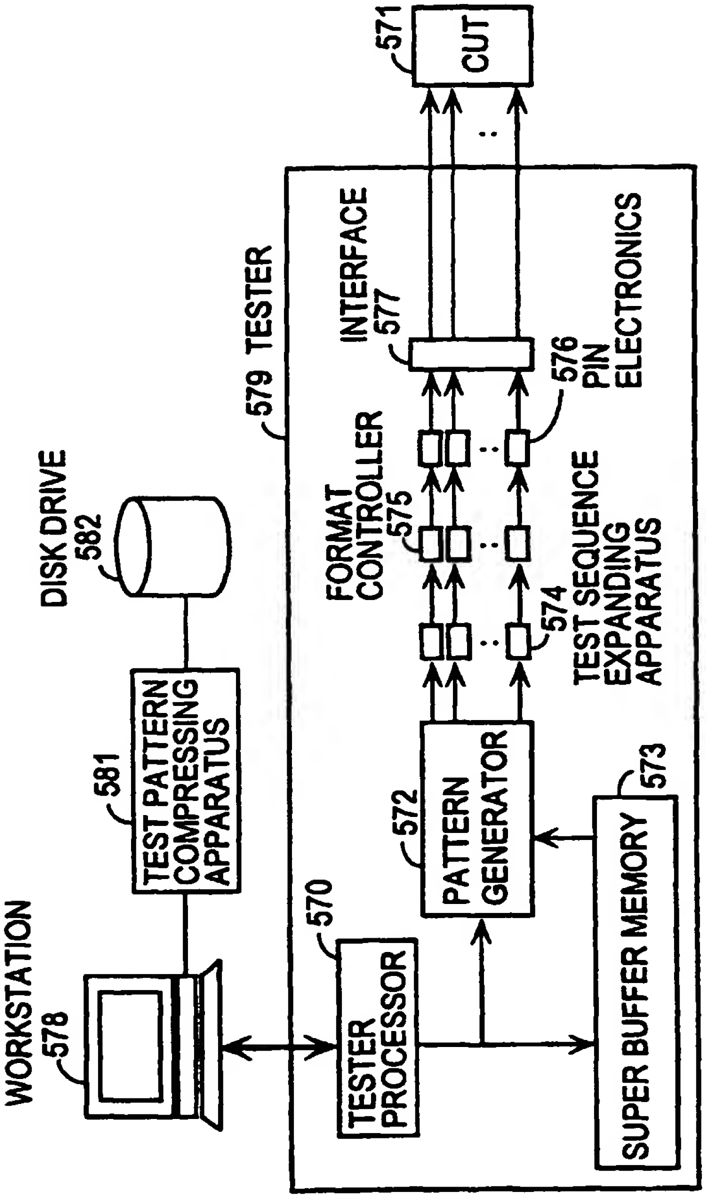

WORKSTATION
578
TEST PATTERN COMPRESSING APPARATUS
581
DISK DRIVE
582
579 TESTER
TESTER PROCESSOR
570
PATTERN GENERATOR
572
SUPER BUFFER MEMORY
573
TEST SEQUENCE EXPANDING APPARATUS
574
FORMAT CONTROLLER
575
576 PIN ELECTRONICS
INTERFACE
577
CUT
571

FIG. 109

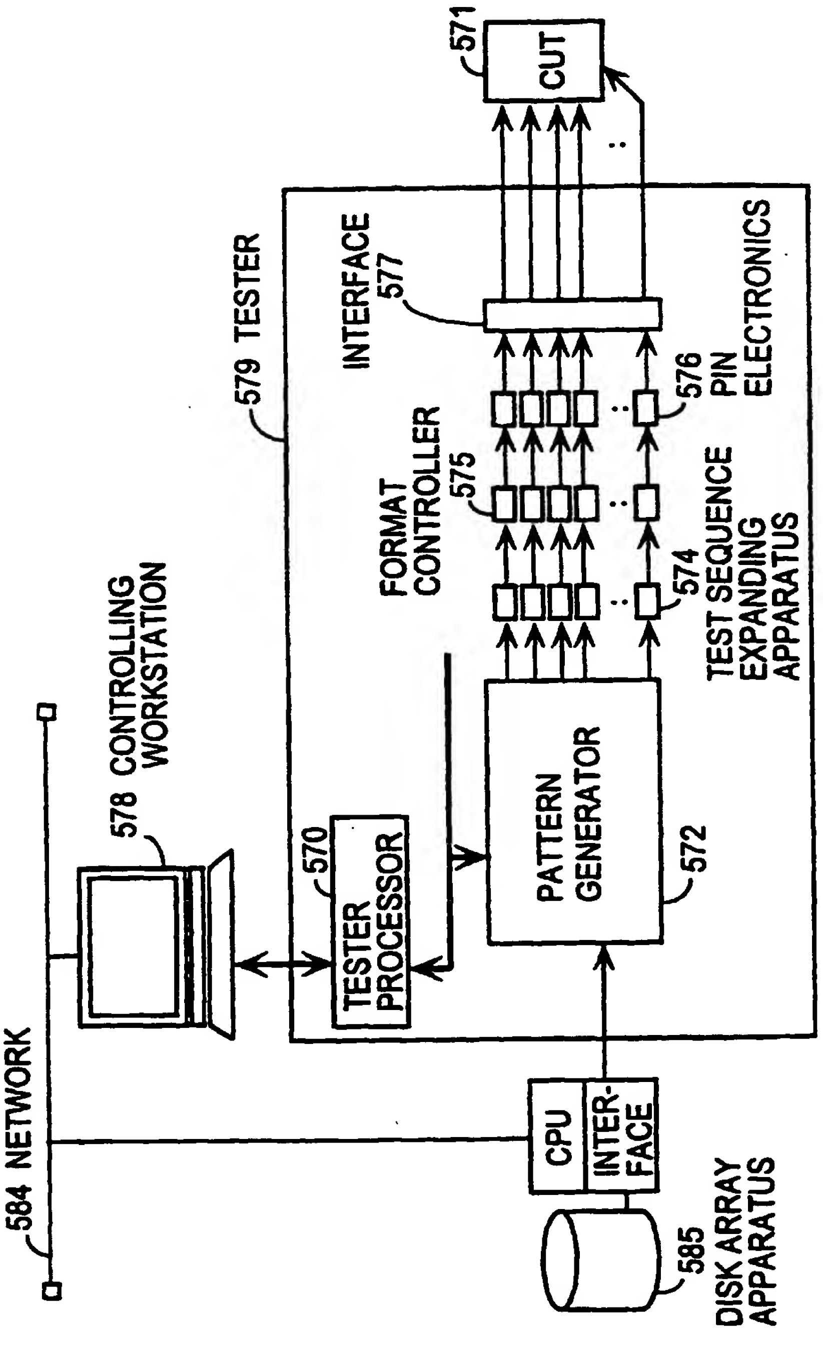

584 NETWORK
578 CONTROLLING WORKSTATION
579 TESTER
570 TESTER PROCESSOR
PATTERN GENERATOR
572
574
TEST SEQUENCE EXPANDING APPARATUS
FORMAT CONTROLLER
575
576 PIN ELECTRONICS
INTERFACE 577
571 CUT
CPU
INTER-FACE
585 DISK ARRAY APPARATUS

FIG. 110

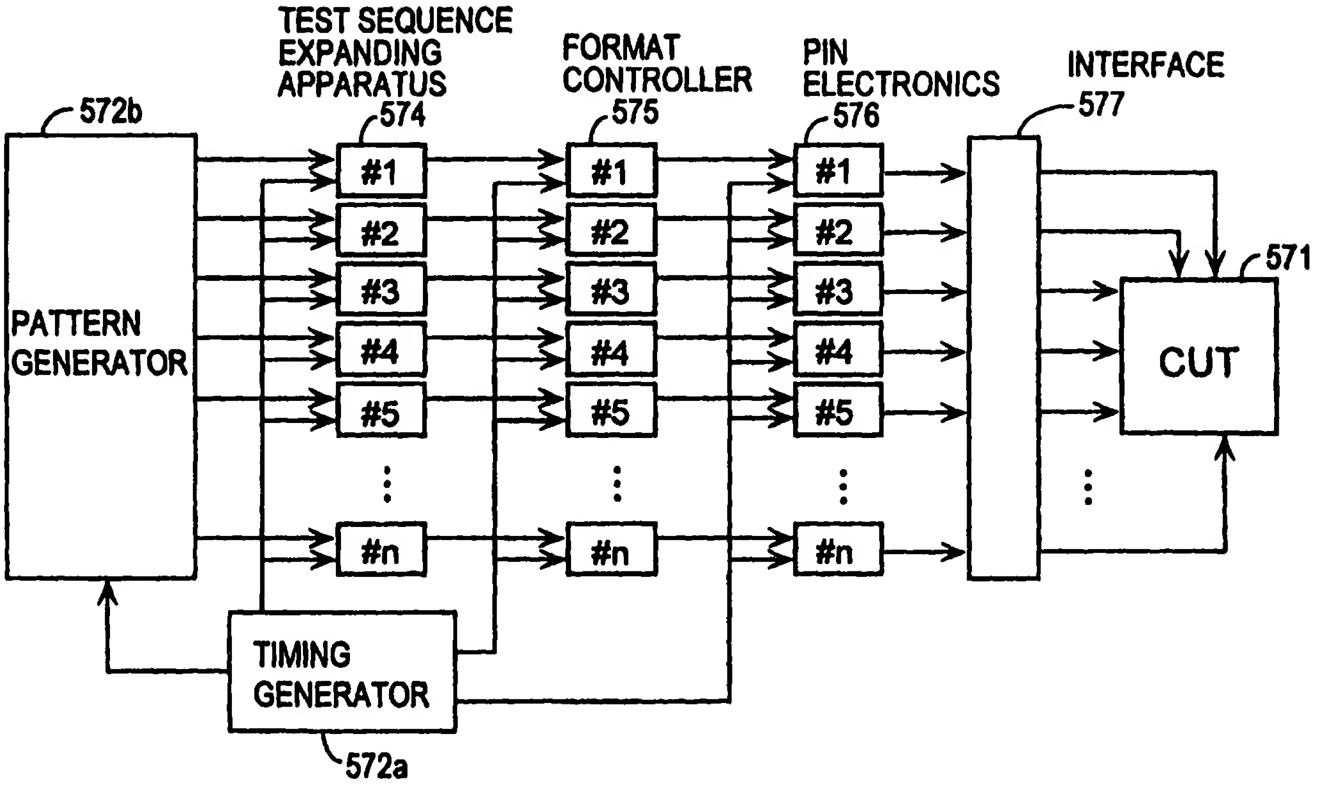

TEST SEQUENCE EXPANDING APPARATUS
574
FORMAT CONTROLLER
575
PIN ELECTRONICS
576
INTERFACE
577
572b
PATTERN GENERATOR
#1
#2
#3
#4
#5
#n
CUT
571
TIMING GENERATOR
572a

FIG. 111

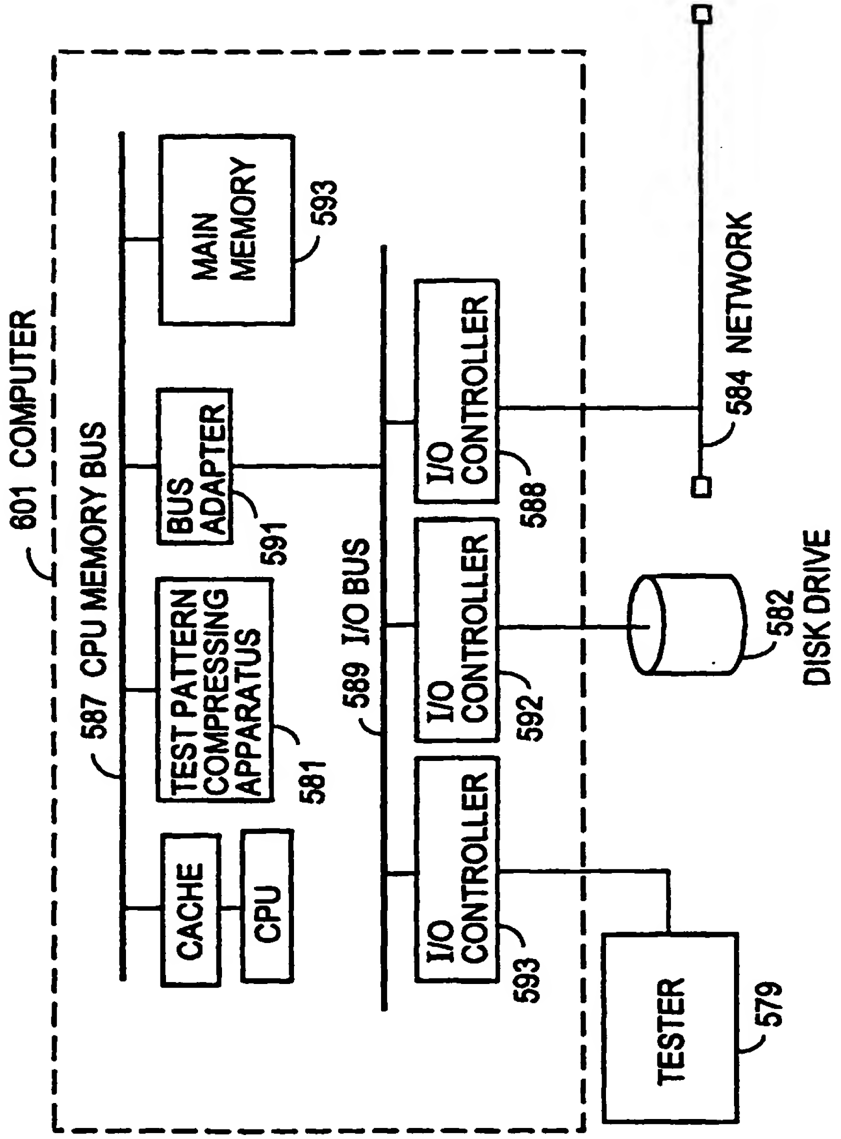

601 COMPUTER
587 CPU MEMORY BUS
CACHE
CPU
TEST PATTERN COMPRESSING APPARATUS
581
BUS ADAPTER
591
MAIN MEMORY
593
589 I/O BUS
I/O CONTROLLER
593
I/O CONTROLLER
592
I/O CONTROLLER
588
TESTER
579
582
DISK DRIVE
584 NETWORK

FIG. 112

FIG. 113

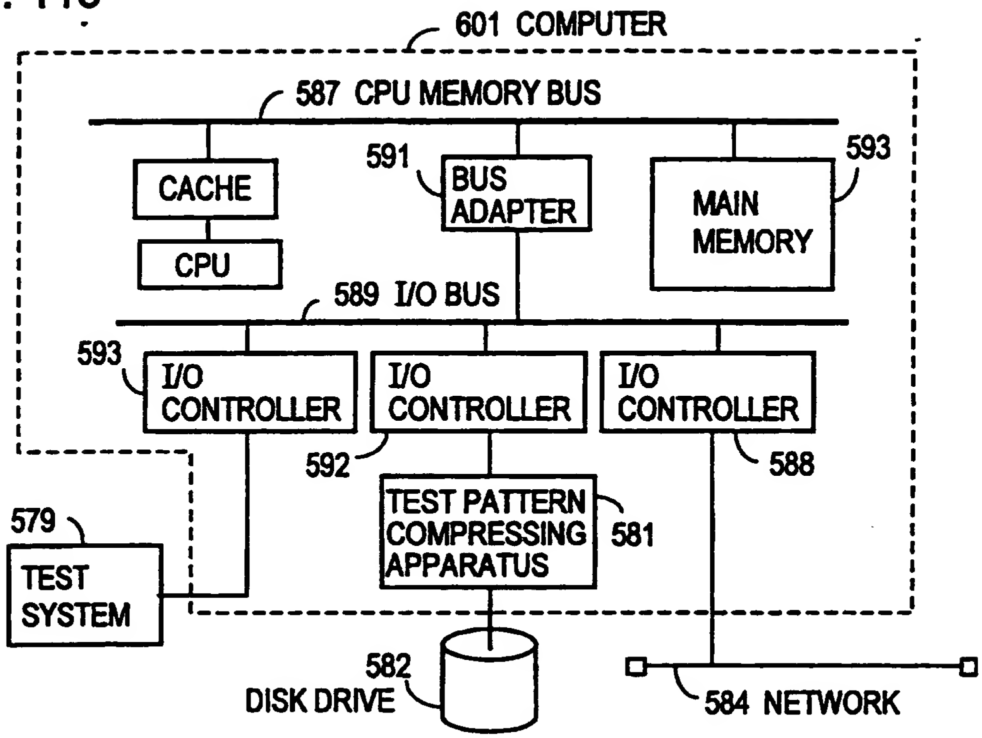

601 COMPUTER
587 CPU MEMORY BUS
CACHE
CPU
591
BUS ADAPTER
593
MAIN MEMORY
589 I/O BUS
593
I/O CONTROLLER
I/O CONTROLLER
I/O CONTROLLER
592
588
579
TEST SYSTEM
TEST PATTERN COMPRESSING APPARATUS
581
582
DISK DRIVE
584 NETWORK

FIG. 114

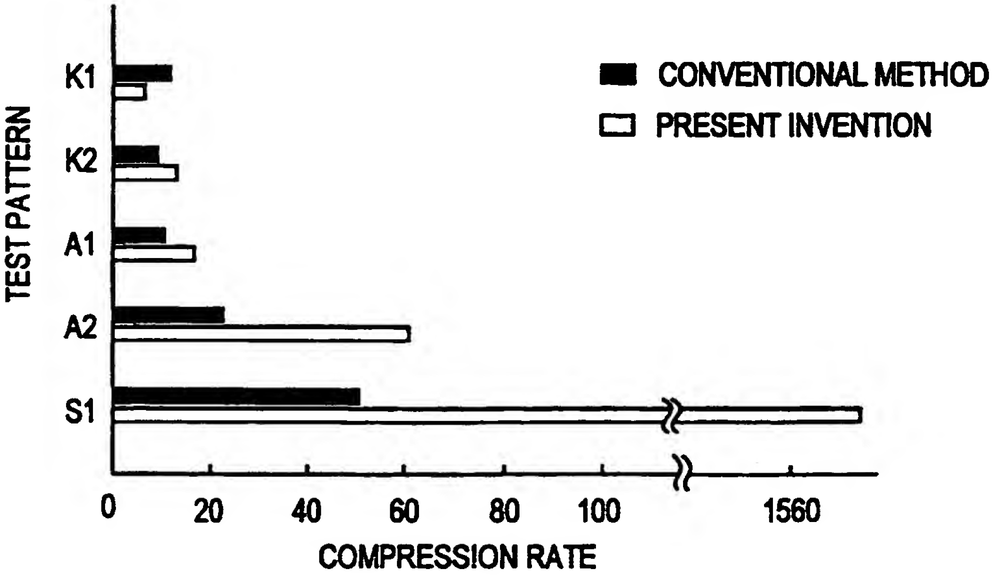


FIG. 115

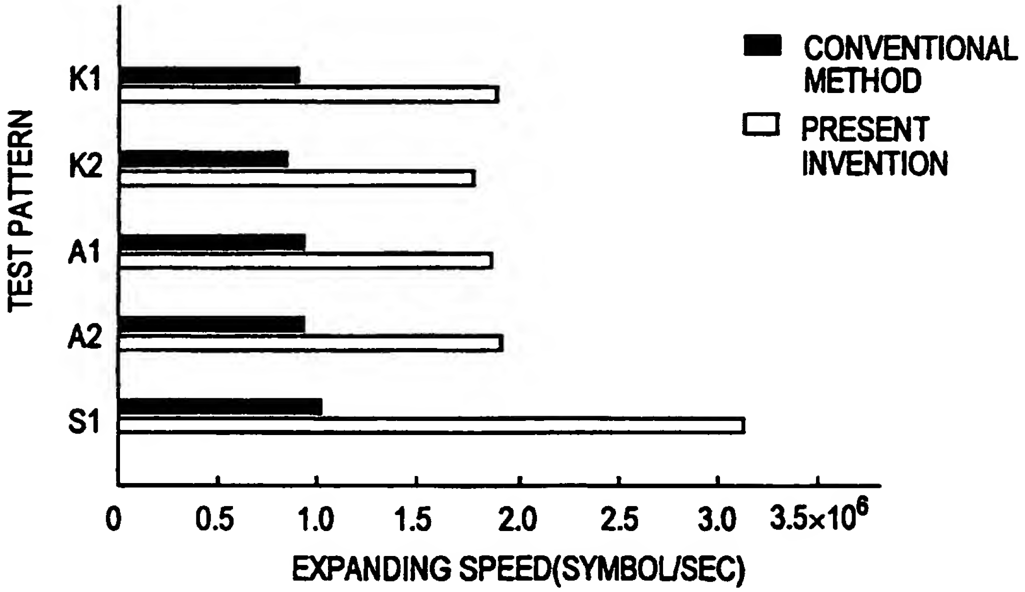
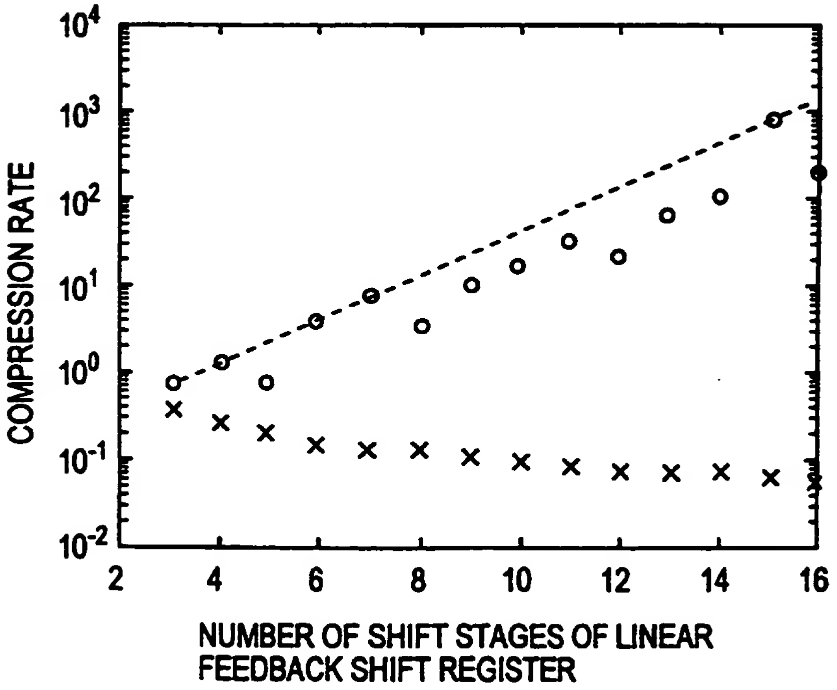


FIG. 116



INTERNATIONAL SEARCH REPORT

International application No.
PCT/JP98/01273

A. CLASSIFICATION OF SUBJECT MATTER
Int.Cl^6 H03M7/40, H03M7/46, G01R31/3183

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
Int.Cl^6 H03M7/40, H03M7/46

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Jitsuyo Shinan Koho	1926-1998	Toroku Jitsuyo Shinan Koho	1994-1998
Kokai Jitsuyo Shinan Koho	1971-1998	Jitsuyo Shinan Toroku Koho	1996-1998

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	JP, 62-118642, A (Canon Inc.), May 30, 1987 (30. 05. 87), Refer to Figs. 1 to 4, 12 and their illustration (Family: none)	1, 2, 4, 20, 21, 24
Y		3, 5-19, 22, 23, 25-50
Y	JP, 1-93217, A (Matsushita Electric Industrial Co., Ltd.), April 12, 1989 (12. 04. 89), Refer to Claims (Family: none)	3, 5, 6, 8-18
Y	JP, 2-34038, A (Hitachi, Ltd.), February 5, 1990 (05. 02. 90), Refer to Claims (Family: none)	16-18, 33, 37
Y	P.M. Fenwick, The Computer Journal, Vol. 39, No. 9, 1996 p.731-740 "The Burrows-Wheeler Transform for Block Sorting Text Compression: Principles and Improvements" Refer to full text	7, 14, 17, 18, 22, 27, 31-33, 36, 37, 40, 43, 44

☒ Further documents are listed in the continuation of Box C. ☐ See patent family annex.

* Special categories of cited documents:
"A" document defining the general state of the art which is not considered to be of particular relevance
"E" earlier document but published on or after the international filing date
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
"O" document referring to an oral disclosure, use, exhibition or other means
"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"&" document member of the same patent family

Date of the actual completion of the international search June 16, 1998 (16. 06. 98)	Date of mailing of the international search report June 30, 1998 (30. 06. 98)
Name and mailing address of the ISA/ Japanese Patent Office	Authorized officer
Facsimile No.	Telephone No.

Form PCT/ISA/210 (second sheet) (July 1992)

INTERNATIONAL SEARCH REPORT

International application No.
PCT/JP98/01273

C (Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	J. Ziv, A. Lempel, IEEE Transactions on Information Theory, Vol. IT-23, No. 3, 1977 p.337-343 "A Universal Algorithm for Sequential Data Compression" Refer to full text	11, 18, 23, 25, 36, 37
Y	J. Ziv, A. Lempel, IEEE Transactions on Information Theory, Vol. IT-24, No. 5, 1978 p.530-536 "Compression of Individual Sequences via Variable-Rate Coding" Refer to full text	11, 18, 23, 25, 36, 37
Y	Electronics, MAR. 1996, p.52-53, Omusha Refer to full text	12, 18, 26, 36, 37
Y	JP, 8-146088, A (Teladain Inc.), June 7, 1996 (07. 06. 96), Refer to Claims & US, 5581177, A & DE, 4423186, A & GB, 2280963, A	19, 28-50
A	H. Yokoo, M. Takahashi, IEICE Trans. Fundamentals, Vol. E79-A, No. 5 May 1996 p.681-686 "Data Compression by Context Sorting" Refer to full text	7, 14, 17, 18, 22, 27, 31-33, 36, 37, 40, 43, 44
A	JP, 61-107818, A (NEC Corp.), May 26, 1986 (26. 05. 86) & US, 4706265, A & EP, 180469, A	1-50
A	JP, 62-82723, A (Canon Inc.), April 16, 1987 (16. 04. 87) (Family: none)	1-50
A	JP, 62-98921, A (Canon Inc.), May 8, 1987 (08. 05. 87) (Family: none)	1-50
A	JP, 62-98919, A (Canon Inc.), May 8, 1987 (08. 05. 87) (Family: none)	1-50
A	JP, 61-107818, A (NEC Corp.), May 26, 1986 (26. 05. 86) (Family: none)	1-50
A	JP, 61-3569, A (Nippon Denki Kanji System K.K.), January 9, 1986 (09. 01. 86) (Family: none)	1-50

Form PCT/ISA/210 (continuation of second sheet) (July 1992)